

AD-A259 023



①

AFIT/GE/ENG/92D-37

DETECTION OF SPOOFING, JAMMING, OR FAILURE
OF A GLOBAL POSITIONING SYSTEM (GPS)

THESIS

Juan R. Vasquez
Captain, USAF

AFIT/GE/ENG/92D-37



93-00075



171
195

012225

Approved for public release; distribution unlimited

DETECTION OF SPOOFING, JAMMING, OR FAILURE
OF A GLOBAL POSITIONING SYSTEM (GPS)

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

Juan R. Vasquez, B.S. Electrical Engineering
Captain, USAF

December, 1992

Approved for public release; distribution unlimited

Preface

This thesis is the result of hard work and dedicated research. It is not the product of any one individual but is a culmination of the efforts of several people. It was sponsored by the Central Inertial Guidance Test Facility (CIGTF), Holloman AFB, NM who are noted for their contributions to the Air Force's pursuit of well integrated navigation systems. The research goals approached by this thesis are aimed at military applications in a wartime situation, but I hope the fundamental concepts that were proved can contribute to many applications in the area of improved navigation.

Although this thesis was not strictly defined as a follow-on to the work done by Captains Negast, Solomon, Snodgrass, and Stacey, I could not have completed my research without extensive use of the computer software they developed. I owe them a great deal of thanks and recognition. I especially owe my gratitude to the now Mr Bill Negast who practically coached me through the early stages of software development. I wish him and his family good luck and success.

I would also like to thank Mr Donald Smith who kept the SPARC 2 computers "alive" when no one else could. His ability to make these powerful but often temperamental machines crunch on data, generate quality printouts, and somehow create more and more disk space is noteworthy. None of us could have completed our work without his efforts.

Next, a personal and professional appreciation for the members on my thesis committee. To the newly promoted Lieutenant Colonel Robert Riggins, my advisor, I owe a big "thanks". He provided me with guidance and inspiration, but his good-natured attitude and understanding made the hard work seem easier. I was extremely lucky to have worked with him. Dr Peter Maybeck is credited with ensuring my thesis would be a quality product. His extensive knowledge and uncanny attention to detail prevented countless mistakes from becoming typeset in this document. I also appreciate the moral support he provided me at work and at home. Captain Dennis Ruck prevented me from waiting until the last minute to start writing and ensured I expressed ideas in such a way that someone other than myself could understand them. I thank you all.

Finally, a combination of love and gratitude to my wonderful fiancé Angela. She not only endured periods of neglect allowing me the time to do my work, but she kept the dream of my other thesis (our wedding) alive. My graduation from AFIT will fill me with pride, but becoming her husband two weeks later will somehow overshadow everything else I have done. I love you.

Juan R. Vasquez

DTIC QUALITY INSPECTED 1

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

Table of Contents

	Page
Preface	ii
Table of Contents	iv
List of Tables	viii
List of Symbols	ix
Abstract	xv
 I. Introduction	 1-1
1.1 Background	1-1
1.2 Problem Definition	1-2
1.3 Scope	1-3
1.4 Assumptions and System Description	1-4
1.5 Literature Review	1-7
1.5.1 Simple Error Detection.	1-8
1.5.2 Direct and Analytic Redundancy.	1-8
1.5.3 Multiple Generalized Likelihood Ratio Testing.	1-9
1.5.4 Chi-Square Testing.	1-10
1.5.5 Multiple Model Adaptive Estimation.	1-11
1.5.6 Literature Review Conclusion.	1-11
1.6 Methodology	1-13
1.6.1 Preliminary Studies.	1-13
1.6.2 System Tests.	1-14
1.6.3 Adaptive Techniques.	1-14
1.6.4 Stopping Criteria.	1-14
1.7 Overview of Thesis	1-15

	Page
II. Kalman Filtering and Failure Detection	2-1
2.1 Overview	2-1
2.2 The Extended Kalman Filter	2-1
2.2.1 The Sampled Data Kalman Filter.	2-1
2.2.2 The Discrete-Time Kalman Filter.	2-4
2.3 Failure Detection	2-5
2.3.1 GLR Equations.	2-5
2.3.2 Chi-Square Equations.	2-8
2.3.3 Threshold Selection and Filter Tuning.	2-9
2.4 Summary	2-10
III. Navigation and Failure Models	3-1
3.1 Overview	3-1
3.2 Overall System Description	3-1
3.3 Model Descriptions	3-1
3.3.1 INS Model.	3-2
3.3.2 The 26-State RRS Error Model.	3-5
3.3.3 The 30-State GPS Error Model.	3-9
3.4 Failure Models	3-15
3.4.1 Simulation Failure Models.	3-15
3.4.2 FDI Failure Models.	3-18
3.5 Summary	3-20
IV. Results and Analysis	4-1
4.1 Assumption Verification	4-1
4.2 NRS Filter Performance	4-1
4.3 FDI Performance	4-3
4.3.1 Jamming Detection.	4-4

	Page
4.3.2 Bias Failures.	4-5
4.3.3 Ramp Failures.	4-8
4.3.4 Failed GPS.	4-9
4.3.5 Numerical Precision and Modelling.	4-10
4.4 Summary	4-11
V. Conclusions and Recommendations	5-1
5.1 NRS Filter Performance	5-1
5.2 FDI Performance	5-1
5.2.1 Corrective Feedback.	5-3
5.3 Recommendations	5-4
Appendix A. Error Model State Definitions	A-1
Appendix B. Dynamics Matrices and Noise Values	B-1
B.1 Definition of Dynamics Matrices	B-1
B.2 Elements of the Process Noise and Measurement Noise Matrices	B-4
Appendix C. Miscellaneous Plots	C-1
Appendix D. Baseline Filter Plots	D-1
Appendix E. Jamming Failure Plots	E-1
Appendix F. Bias Failure Plots	F-1
Appendix G. Ramp Failure Plots	G-1
Appendix H. GPS Failure Plots	H-1
Appendix I. Software	I-1
Bibliography	BIB-1

	Page
Vita	VITA-1

List of Figures

Figure	Page
1.1. Jamming and Spoofing	1-2
1.2. Types of Failures	1-3
1.3. NRS Block Diagram	1-5
1.4. Multiple GLR Testing	1-10
1.5. Multiple Model Adaptive Estimation	1-12
3.1. Truth and Filter Model Block Diagram	3-2
3.2. RRS Measurements	3-9
3.3. GPS Measurements	3-13
C.1. Two Hour Flight Profile	C-2
C.2. Element of the State Dynamics Matrix \mathbf{F}	C-3
C.3. Spoofers Induced Errors	C-4
D.1. State Plots: Baseline	D-3
D.2. State Plots: Baseline	D-4
D.3. State Plots: Baseline	D-5
D.4. State Plots: Baseline	D-6
D.5. State Plots: Baseline	D-7
D.6. State Plots: Baseline	D-8
D.7. State Plots: Baseline	D-9
D.8. State Plots: Baseline	D-10
D.9. State Plots: Baseline	D-11
D.10.State Plots: Baseline	D-12
D.11.State Plots: Baseline	D-13
D.12.State Plots: Baseline	D-14

Figure	Page
D.13.State Plots: Baseline	D-15
D.14.State Plots: Baseline	D-16
D.15.State Plots: Baseline	D-17
D.16.Residual Plots: Baseline	D-18
D.17.Residual Plots: Baseline	D-19
D.18.Residual Plots: Baseline	D-20
D.19.Residual Plots: Baseline	D-21
D.20.Residual Plots: Baseline	D-22
D.21.GLR/CHI: Baseline	D-23
E.1. State Plots: Heavy Jamming	E-2
E.2. State Plots: Heavy Jamming	E-3
E.3. State Plots: Heavy Jamming	E-4
E.4. Residual Plots: Heavy Jamming	E-5
E.5. Residual Plots: Heavy Jamming	E-6
E.6. Residual Plots: Heavy Jamming	E-7
E.7. GLR: Heavy Jamming	E-8
E.8. Fail Flag: Heavy Jamming	E-9
E.9. CHI: Heavy Jamming	E-10
E.10.GLR/CHI: Medium Jamming	E-11
E.11.GLR/CHI: Light Jamming	E-12
F.1. State Plots: Run 4, Bias=7000	F-2
F.2. State Plots: Run 4, Bias=7000	F-3
F.3. Residual Plots: Run 4, Bias=7000	F-4
F.4. Residual Plots: Run 4, Bias=7000	F-5
F.5. GLR: Run 4, Bias=7000	F-6
F.6. GLR: Run 4, Bias=7000	F-7

Figure	Page
F.7. Fail Flag: Run 4, Bias=7000	F-8
F.8. MLE $\hat{\nu}(t_k)$: Run 4, Bias=7000	F-9
F.9. CHI: Run 4, Bias=7000	F-10
F.10. GLR/Fail Flag: Run 5, Bias=700, 2000-4000 sec	F-11
F.11. CHI: Run 5, Bias=700, 2000-4000 sec	F-12
F.12. GLR: Run 6, Bias=700, 4000-6000 sec	F-13
F.13. Fail Flag: Run 6, Bias=700, 4000-6000 sec	F-14
F.14. CHI: Run 6, Bias=700, 4000-6000 sec	F-15
F.15. State Plots: Run 6, Bias=700 on All	F-16
F.16. State Plots: Run 7, Bias=700 on All	F-17
F.17. State Plots: Run 7, Bias=700 on All	F-18
F.18. GLR/Fail Flag: Run 7, Bias=700 on All	F-19
G.1. State Plots: Run 8, Ramp=2T	G-2
G.2. State Plots: Run 8, Ramp=2T	G-3
G.3. Residual Plots: Run 8, Ramp=2T	G-4
G.4. Residual Plots: Run 8, Ramp=2T	G-5
G.5. GLR: Run 8, Ramp=2T	G-6
G.6. GLR: Run 8, Ramp=2T	G-7
G.7. Fail Flag: Run 8, Ramp=2T	G-8
G.8. CHI: Run 8, Ramp=2T	G-9
G.9. GLR: Run 9, Ramp=1T	G-10
G.10. Fail Flag: Run 9, Ramp=1T	G-11
G.11. CHI: Run 9, Ramp=1T	G-12
H.1. State Plots: Run 10, GPS Fail	H-2
H.2. Residual Plots: Run 10, GPS Fail	H-3
H.3. Residual Plots: Run 10, GPS Fail	H-4

List of Tables

Table	Page
1.1. Orbit Runs	1-14
3.1. Failure Types and Models	3-16
4.1. Temporal Averages of True Filter Errors (1σ)	4-2
A.1. 41-State INS System Model: First 20 States	A-2
A.2. 41-State INS System Model: Second 21 States	A-3
A.3. 26-State RRS System Model	A-4
A.4. 30-State GPS System Model	A-5
A.5. 15-State Reduced-Order Filter Model	A-6
B.1. Elements of the Dynamics Submatrix \mathbf{F}_{11}	B-2
B.2. Elements of the Dynamics Submatrix \mathbf{F}_{12}	B-2
B.3. Elements of the Dynamics Submatrix \mathbf{F}_{13}	B-3
B.4. Elements of the Dynamics Submatrix \mathbf{F}_{14}	B-3
B.5. Elements of the Dynamics Submatrix \mathbf{F}_{22}	B-4
B.6. Elements of the Dynamics Matrix for GPS & RRS	B-4
B.7. Elements of Truth Model Process Noise Submatrix \mathbf{Q}_{11}	B-5
B.8. Elements of Truth Model Process Noise Submatrix \mathbf{Q}_{22}	B-5
B.9. Elements of Truth Model Process Noise for GPS & RRS	B-5
B.10. Filter Process Noise \mathbf{Q}	B-5
B.11. Truth and Filter Measurement Noises \mathbf{R}	B-5

List of Symbols

Symbol		Page
$\mathbf{f}[\mathbf{x}(t), t]$	nonlinear state dynamics vector	2-1
$\mathbf{x}(t)$	state vector	2-1
t	time	2-1
$\mathbf{G}(t)$	process noise input matrix	2-1
\mathbf{I}	identity matrix	2-1
$\mathbf{w}(t)$	process noise	2-1
$\mathbf{Q}(t)$	process noise strength	2-1
$\mathbf{z}(t_i)$	discrete measurement vector	2-2
$\mathbf{v}(t_i)$	measurement noise	2-2
$\mathbf{R}(t_i)$	measurement noise strength	2-2
$\mathbf{h}[\mathbf{x}(t_i), t_i]$	nonlinear observation vector	2-2
$\mathbf{x}_n(t)$	nominal state vector	2-2
$\delta \mathbf{x}(t)$	perturbation state vector	2-2
$\mathbf{F}[t; \mathbf{x}_n(t)]$	linearized state dynamics matrix	2-2
$\mathbf{H}[t_i; \mathbf{x}_n(t_i)]$	linearized observation matrix	2-3
$\widehat{\delta \mathbf{x}}(t)$	error state estimate vector	2-3
$\hat{\mathbf{x}}(t)$	state estimate vector	2-3
$\mathbf{P}(t/t_i)$	state covariance matrix	2-3
$\mathbf{K}(t_i)$	Kalman filter gain	2-4
$\Phi(t_i, t_{i-1})$	state transition matrix	2-4
$\mathbf{d}(t_i)$	failure vector	2-5
$n(t_i, \theta)$	failure function	2-5
ν	unknown size of the failure	2-5
θ	unknown time of failure	2-5
$l(t_i, \theta)$	likelihood function	2-6

Symbol		Page
$\gamma(t_i)$	measurement residuals	2-7
$\mathbf{m}(t_i, \theta)$	failure residual offset	2-7
$\mathbf{y}(t_{i+1}, \theta)$	recursive failure quantity	2-7
$\Lambda(t_j)$	residual covariance	2-8
$S(t_i, \theta)$	linear combination of residuals	2-8
$C(t_i, \theta)$	deterministic failure value	2-8
ϵ	failure threshold	2-8
$\chi(t_k)$	chi-square random variable	2-9
$\delta \mathbf{x}_1$	general INS errors	3-3
$\delta \mathbf{x}_2$	time-correlated INS errors	3-3
$\delta \mathbf{x}_3$	gyro bias errors	3-3
$\delta \mathbf{x}_4$	accelerometer bias errors	3-3
$\delta \mathbf{x}_5$	thermal transient errors	3-3
$\delta \mathbf{x}_6$	gyro compliance errors	3-3
h_t	true altitude	3-4
δh_B	total barometric altimeter error	3-4
δh	error in vehicle altitude above reference ellipsoid	3-4
δV_{t_i}	truth state velocity error	3-4
δV_i	filter state velocity error	3-4
x_{br}	RRS range bias	3-5
x_{bv}	RRS velocity bias	3-5
R_{RRS}	RRS range measurement	3-7
R_t	true range	3-7
δR_{atm}	atmospheric delay error	3-7
δR_b	equipment calibration error	3-7
R_{INS}	range to INS	3-7
\mathbf{X}_U	user position vector	3-8

Symbol		Page
\mathbf{X}_T	transponder position vector	3-8
x_{Uclk_b}	user clock bias	3-10
$x_{Uclk_{dr}}$	user clock drift	3-10
R_{GPS}	GPS pseudorange measurement	3-12
δR_{cl}	code loop error	3-12
δR_{trop}	tropospheric delay	3-12
δR_{ion}	ionospheric delay	3-12
δR_{Sclk}	satellite clock error	3-12
δR_{Uclk}	user clock error	3-12
\mathbf{X}_S	satellite position vector	3-12
RS	jamming noise	3-15
Δ	code modulation chip width	3-15
C/N_0	carrier-to-noise ratio	3-15
B_n	code tracking loop noise bandwidth	3-15
B_i	one-sided IF bandwidth	3-15
K_1	code mechanization parameter constant	3-15
K_2	code mechanization parameter constant	3-15
$\delta\theta_{x:spoof}$	latitude error with spoofing	3-17
$\delta\theta_{y:spoof}$	longitude error with spoofing	3-17
$\mathbf{u}(t)$	control input	5-3

Abstract

The Air Force has equipped its aircraft with avionic systems such as Global Positioning Systems (GPS) and Inertial Guidance Systems (INS) capable of providing accurate navigation solutions. The aircrews flying these aircraft require a system that can either survive the hostile environments encountered in combat or notify the aircrew that their performance has been significantly degraded. This research focuses on failure detection and isolation techniques using an extended Kalman filter and generalized likelihood ratios using matched filters. Analysis is conducted using a Kalman filter development package known as the Multimode Simulation for Optimal Filter Evaluation (MSOFE). Both a large order *truth* model for the navigation system (in which a full 24 satellite constellations is modeled) and a reduced-order Kalman filter are developed. Results suggest that failures within the GPS can be detected, isolated, and in some cases compensated through feedback.

DETECTION OF SPOOFING, JAMMING, OR FAILURE OF A GLOBAL POSITIONING SYSTEM (GPS)

I. Introduction

A variety of Global Positioning System (GPS) receivers and Inertial Navigation Systems (INS) are installed on military aircraft. The GPS receives information from orbiting satellites and calculates estimates of the position and velocity of the aircraft. The INS detects inertial motion of the aircraft and calculates its own estimates of aircraft position and velocity. The GPS and INS are integrated to form a system that is more accurate and reliable than each of these systems by itself. Additional measurements from ground based transponders are available and aid in determining aircraft position. The transponder based system is referred to as the Range/Range-Rate System (RRS). The combination of the GPS, INS, and RRS form the Navigation Reference System (NRS) whose primary function is to assist the pilot in navigating the aircraft. An important note is that the RRS is often used on test ranges to reconstruct flight paths over the range. It is not desired to limit the applications of this thesis to test aircraft, so it is assumed that the RRS represents any one of many ground-based transponder systems used on current military aircraft.

1.1 Background

Research is being done by the Central Inertial Guidance Test Facility (CIGTF), 6585th Test Group, Air Force Systems Command (AFSC), Holloman AFB, NM to determine the vulnerability of the GPS to jamming and spoofing. Jamming is nothing more than bombarding the GPS receiver with electronic noise. Figure 1.1 illustrates jamming as a broad band of noise directed at the GPS receiver. There is no intent to mislead the GPS receiver with jamming, but rather to prevent it from receiving the desired satellite information. Spoofing is more complex and harder to detect. The goal of spoofing is to mimic the signal sent from the satellite to the GPS receiver but with minor changes to the signal. These minor changes will cause the GPS to calculate erroneous estimates and draw

the aircraft away from its desired destination. Spoofing is shown in Figure 1.1 as a directed signal from a ground-based platform aimed at a particular target. An airborne platform for the spoofer would be more effective against aircraft with directional GPS antennas capable of pointing away from a ground-based spoofer. Jamming and spoofing constitute two of the three failures that will be addressed in this thesis. A third failure that can affect the performance of the GPS is the loss of a pseudorange input to the receiver. Failures caused by losing a GPS pseudorange may look similar to spoofing or jamming, but it is important to identify which of the three failures has occurred.

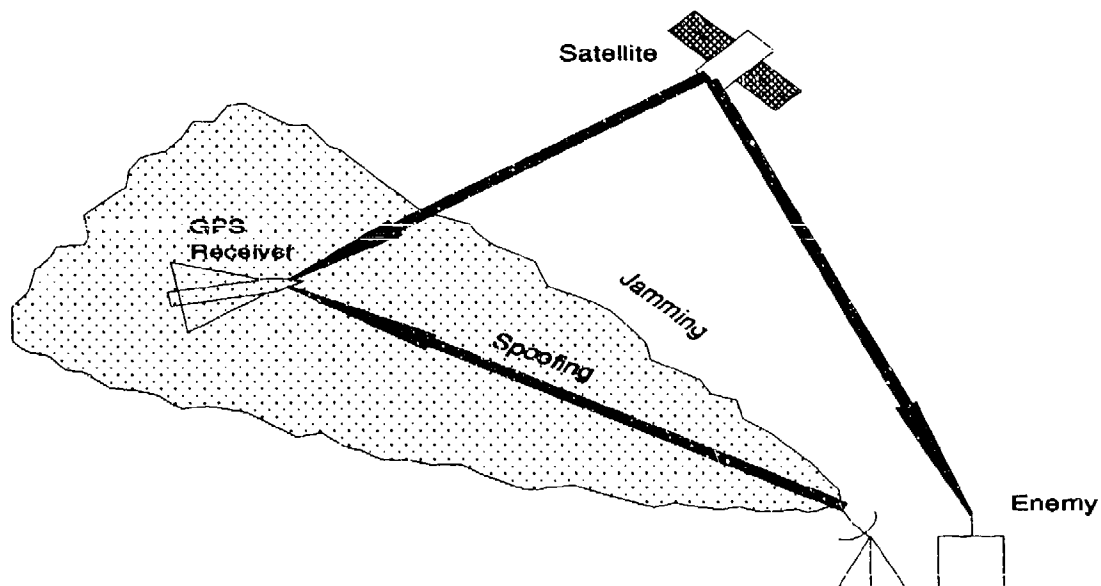


Figure 1.1. Jamming and Spoofing

1.2 Problem Definition

The primary goal of this thesis will be to develop a failure detection and isolation (FDI) system that will identify GPS failures as described above. This FDI system will be based on techniques discussed in the literature review. The follow-on goal will be to develop an adaptive system which can correct for a failure that has been detected and isolated. The NRS is to retain good performance as a result of corrections which are typically used in a feedback configuration.

1.3 Scope

Two types of failures will be considered. Figure 1.2 depicts the two types of failures as a jump or a ramp. A jump failure results in a corrupted signal with a different magnitude than the desired signal and occurs almost instantaneously in time. Some FDI algorithms assume the magnitude of this jump is known but it is more realistic to assume the magnitude is unknown and important for the FDI system to determine. A ramp failure increases more gradually than a jump. The deceiving signal is constantly increasing away from the desired signal at an unknown rate. The thresholds and delay shown will be discussed in the literature review. The rate of increase of the ramp and magnitude of the jump will be varied for different studies.

Although the failure algorithms discussed in this thesis could apply to systems other than the GPS, no attempt will be made to induce failures into the INS or RRS, as the primary focus will be on the GPS only. Criteria for completion of the research are presented in Section 1.6.

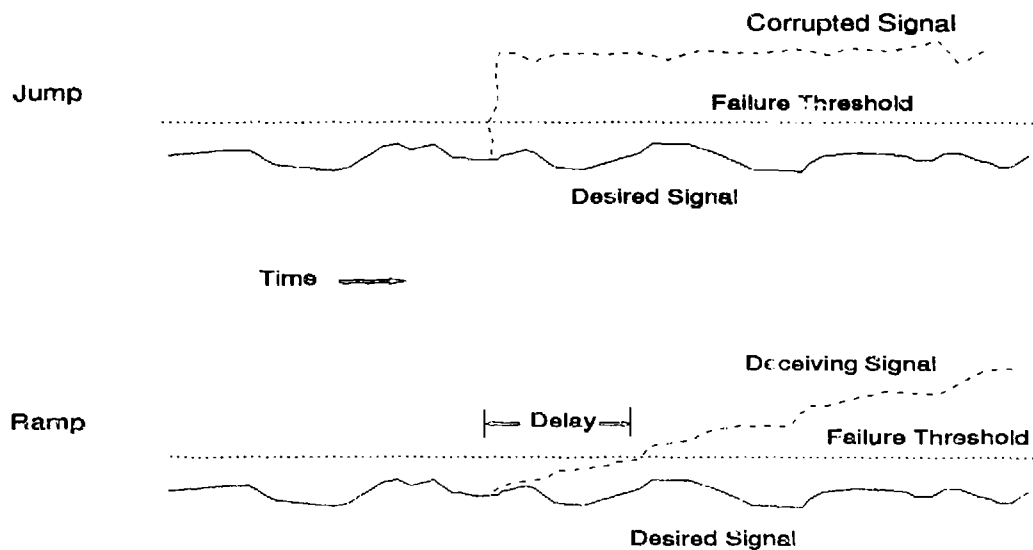


Figure 1.2. Types of Failures

1.4 Assumptions and System Description

The list that follows helps describe the scenario chosen for this thesis and the simplifying assumptions that were adopted. The impact caused by these assumptions may not be apparent without knowing the context in which they were applied, so references are listed to help the reader see their significance.

1. The NRS will be mounted on a computer-simulated aircraft that is capable of high dynamic maneuvers analogous to a military fighter jet. Actual flight tests will not be performed, but a two-hour flight profile as shown in Figure C.1 will be simulated on a computer. This profile is represented by the latitude, longitude, and altitude of the aircraft.
2. A typical NRS configuration will be used as shown by a block diagram representation in Figure 1.3. Error states are used as the basis of the filter design model, and difference measurements are provided to a Kalman filter by the GPS, INS, and RRS. The Kalman filter generates error state estimates used to correct the original INS states, resulting in refined estimates of latitude, longitude, and altitude. There are fourteen measurements available to the filter, including four satellite pseudoranges, six transponder ranges, velocity in 3 axes via Doppler aiding and altitude from the barometric altimeter. Chapter III provides a detailed description of these measurement sources. Residuals are sent from the Kalman filter to the FDI system and corrections can be fed back to the filter. The INS is inherently unstable in the altitude channel and receives stability aiding from a barometric altimeter (1:p. 83). A sampling period of two seconds (sec) was chosen for all the measurements. Previous AFIT research (23, 26, 27) used update periods ranging from two to ten seconds. The slower sampling periods were typically chosen to speed up the simulations which took several days to run. This thesis used truth and filter models with fewer states resulting in shorter simulations so a higher sampling period was feasible. However, data files were used to store several values at each sample time and these files became unmanageable with sample periods below two seconds. A few simulations were also

conducted with a one second sampling period to ensure that no significant losses occurred in the filter and FDI performance.

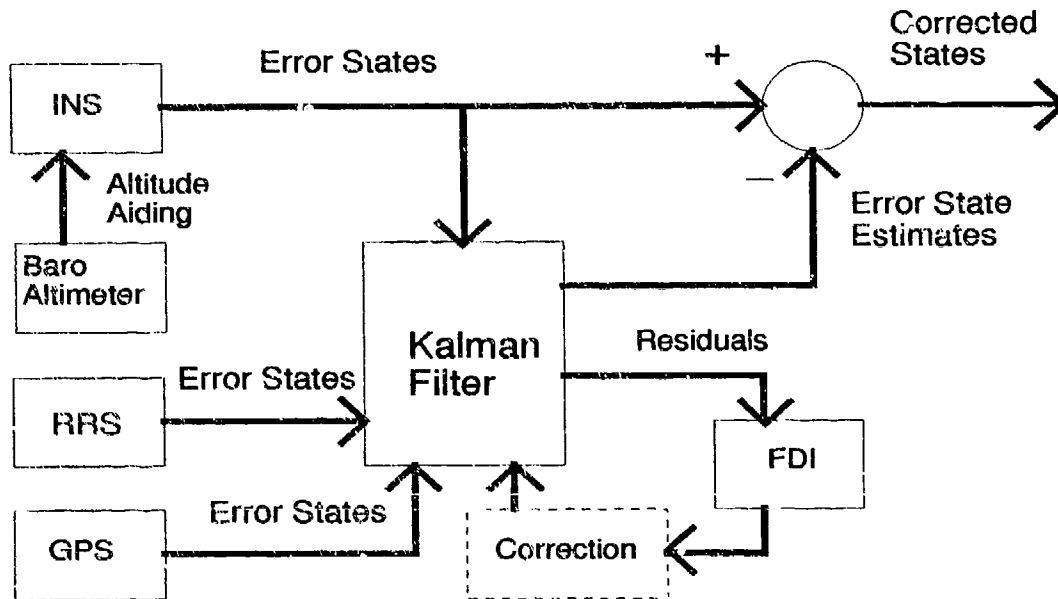


Figure 1.3. NRS Block Diagram

3. The NRS will be modeled using differential equations that describe the physical relationships between the real world and the NRS electronics. The model will be based on error states rather than actual states because error states will provide more accurate estimates of position and velocity (2). Research at CIGTF is based on a Litton LN-93 model for the INS (9), a generic GPS model (23), and a simplified RRS model (23). The models used for this thesis are fully described in Chapter III.
4. Failures will be assumed additive rather than multiplicative, allowing them to be represented as an additional term in the equations describing the measurements from the GPS rather than changes in the dynamics model. This assumption simplifies the problem significantly without a loss of realism. Finally, it will be assumed that

only one failure can occur at a time and that multiple failures are unmodeled. See Sections 1.6.1 and 3.4.

5. Computer simulations will be run using a program called Multimode Simulation for Optimal Filter Evaluation (MSOFE) (22). MSOFE is specifically used to provide analysis of designs involving Kalman filters. A complete analysis, short of actual test flights, is possible using Monte Carlo run simulations with MSOFE. The flight profile will be generated using a computer program called PROFGEN (21). PROFGEN takes user input commands that describe aircraft maneuvers and produces computer-compatible data which is fed to MSOFE for the simulations.
6. The state dynamics matrix \mathbf{F} is considered piecewise constant between sample storage times of two seconds. The discretization process in Matrix_x (8) takes this matrix and uses an 8th order Padé approximation for the matrix exponential $\text{EXP}(\mathbf{F}\Delta t)$. See Sections 2.2.2 and 4.1.
7. The measurements are assumed sufficiently uncorrelated so the off-diagonal terms of the measurement noise matrix, \mathbf{R} , are negligible and can be set to zero. The residual covariance matrix $\mathbf{\Lambda}$ is assumed nearly diagonal (or at least dominated by its diagonal terms) based on the relationship:

$$\mathbf{\Lambda}(j) = \mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i) + \mathbf{R} \quad (1.1)$$

where \mathbf{R} is assumed stationary because it has very small variations throughout the simulation. Although a diagonal \mathbf{R} does not ensure a diagonal $\mathbf{\Lambda}$, Equation (1.1) indicates that observation of the diagonal terms of $\mathbf{\Lambda}$ will provide insights into the condition of the measurements.

8. The simulation and post-processing software (MSOFE and Matrix_x) are coded to run in double precision in order to handle large disparities in various Kalman filter values.
9. MSOFE simulations for filter tuning are performed using 10-run Monte Carlo analyses with statistical values averaged over the 10 runs. Single-run Monte Carlo sim-

ulations are used when testing the failure detection algorithm and will be compared with multi-run results to ensure the single run is not unusual.

10. Failure thresholds will be determined empirically based on simulation results in various situations. See Section 2.3.3.
11. Taylor series truncated to first order will be used for linearizing nonlinear equations. Perturbations about some nominal point will be established in each case. See Sections 2.2.1, 3.3.2.2 and 3.3.3.2.
12. The FDI algorithm will only view the data within a set window of time in order to avoid a growing set of hypotheses, as discussed later in Section 1.5.3. The size of the window will be determined empirically based on simulation results and the window will slide in time to cover all the data in the simulation. For example, a ten-sample window would have new data added to the end of the window and simultaneously have old data deleted from the beginning of the window. Additionally, the failure is assumed to occur at the beginning of the window to simplify calculations. The consequence of this simplification is a delay in detecting the failure caused by waiting for the failure to reach the beginning of the window, as discussed in Section 2.3.1.
13. A Doppler system is available to provide velocity aiding to the INS. The measurements from the Doppler are ideal and tell the filter the exact error between the filter state and the truth state. This ideal situation was assumed to allow direct comparison of results against those obtained in previous AFIT theses. See Section 3.3.1.3.

1.5 Literature Review

This section contains a review of literature pertaining to four FDI techniques with specific interest on their application to the integrated NRS described in Section 1.4. More details on this NRS can be found in a variety of sources, including Air Force Institute of Technology (AFIT) theses (23, 27). References made to the Kalman filter and FDI system will directly apply to the components shown in Figure 1.3. Kalman filter theory

is presented by Maybeck (13). The literature review conclusion will discuss the validity of each technique for use in the thesis research.

1.5.1 Simple Error Detection. A simple method for error detection is to monitor the errors in the position and velocity calculations. A failure has occurred if these errors exceed an established failure threshold, as shown in Figure 1.2. The problem with this method is that the desired values of true position and velocity are not known, so it is not possible to determine the inaccuracy of the calculations. An alternative is to use the Kalman filter to provide a statistic known as code loop pseudorange tracking error which is related to the desired position through differential equations (24:pp. 1460-1461). The tracking error is sent to the FDI system for comparison against a failure level. The failure level will be determined based on past experience, flight tests, and most commonly through computer-generated Monte Carlo simulations or covariance analyses (11:pp. 102-106). If the code loop error for a given satellite is larger than the failure level, the GPS is considered to have failed or lost lock in the channel associated with that satellite and the GPS will no longer use information from this channel until the satellite has regained lock.

1.5.2 Direct and Analytic Redundancy. One of the simplest and most reliable failure detection techniques is the use of redundant elements for voting. Given a system with triple redundancy, an algorithm can be easily written that will compare the outputs of each element, allowing them to vote on the condition of the other elements. Simply stated, if two of the elements agree on the aircraft's position but the third element provides a totally different value, the latter is considered inadequate to provide accurate information and is removed from the system. Once the third element is removed from the system, the algorithm is unable to isolate a failure. If the two remaining elements disagree, a failure has been detected but not isolated and both elements will have to be removed from the system or the system performance will be degraded. The major disadvantage of direct redundancy is the need for redundant hardware (5:pp. 1-2).

A more sophisticated approach uses analytic redundancy. By using the physical and dynamic relationships between instruments on an aircraft, it is possible to generate multiple sources of the same information mathematically. These sources are used by the FDI

structure through voting techniques. Analytic redundancy avoids the need for excessive hardware, because each instrument is required for specific mission requirements. The surplus of information is also used to provide isolation of the failure without the need for triple redundancy (5:pp. 3-7).

1.5.3 Multiple Generalized Likelihood Ratio Testing. In some situations direct redundancy is not practical. An alternative is to use a Kalman filter as shown in Figure 1.3 to compensate for failures in the NRS and provide accurate estimates of the position and velocity. Figure 1.4 shows how the sensor (GPS, INS, or RRS), the Kalman filter, and the FDI block of Figure 1.3 interact. Three hypotheses are considered with H_0 , H_1 , and H_2 , representing no failure, jump failure and ramp failure respectively. The Kalman filter is designed based on H_0 , and two matching filters are designed based on H_1 and H_2 (29). When considering H_1 and H_2 , design parameters within the matching filters determine what type of failure is being matched, but the magnitude of the jump and rate of the slope do not have to be predetermined and are estimated by the GLR algorithm. The matching filters monitor the residuals provided by the Kalman filter, and each matching filter computes a generalized likelihood ratio (GLR). The GLR's are indications of which hypothesis is most correct. The GLR's use maximum likelihood estimates (MLE) and are compared through test logic to detect and isolate failures. A corrective signal can be fed back to the Kalman filter for adaptation to the failure. In many cases, simply adding or subtracting a bias to the sensor allows the system to continue operation without losing the sensor in question. One advantage of the GLR test over other FDI algorithms is that prior knowledge of the magnitude of the failure is not necessary. Further detail on GLR's and MLE's are presented by Willsky and Jones (30, 31).

The simple hypotheses for failure and no-fail conditions may not provide the robustness needed to detect and adapt to certain failures. Detection of ramp failures is particularly difficult when the ramp rate varies significantly. Figure 1.2 shows the time delay involved in detecting ramp failures. This delay occurs because the deceiving signal is slowly moving away from the desired signal and takes more time to cross the failure threshold. By adding filters designed to match the ramp failure, these longer delays can

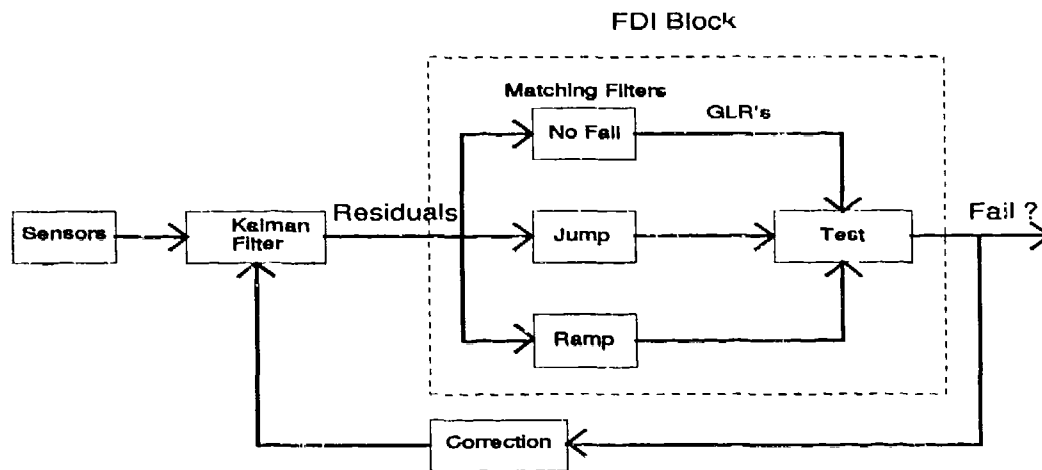


Figure 1.4. Multiple GLR Testing

be avoided. It is also desirable to design the filter based on the time of the failure. If the filter is customized to look for a failure at a specific instant in time, then it has a better chance of good detection. This idea results in a very large bank of filters for long periods of time. The disadvantage of adding filters is the increase in computations required for multiple filters. A solution to this problem uses a set number of filters over a window of time. The number of filters remains constant, and the performance of the FDI system is often maintained (31:pp. 606-608). The justification for using sliding windows is discussed by Willsky (31:pp. 604-605).

1.5.4 Chi-Square Testing. Another FDI method based on residual monitoring is a chi-square test which is similar to GLR testing in that it calculates a random variable $\chi(k)$ based on the filter outputs. One primary difference between these two algorithms is that GLR tests are functions of the dynamics model and chi-square tests are not, as shown mathematically in Section 2.3. A second major difference is that chi-square tests do not try to match the failure and only have one hypothesis, making it a binary test for fail/no-fail. Without the ability to distinguish between types of failures, the chi-square test is not

good for isolation. However, chi-square testing is easy to implement, runs quickly, and can provide the first level of detection in a multi-level FDI scheme.

1.5.5 Multiple Model Adaptive Estimation. A final FDI technique for discussion is the use of multiple models that represent the dynamics of the aircraft under a variety of conditions. Although this technique is analogous to multiple GLR testing in many ways, it differs in its structure and decision making process. Unlike the multiple GLR testing which modeled a variety of failures using matching filters, the multiple model adaptive estimation (MMAE) technique models the dynamic nature of the aircraft and its sensors to represent their behavior in the presence of a failure. Figure 1.5 shows that a separate Kalman filter is designed for each failure condition, and the residuals are used to determine which filter best models the aircraft and its sensors at the current time. The Kalman filter for the jump or ramp might actually consist of many filters designed for various magnitudes and slopes. A probability of accuracy ranging from zero to one is computed for each filter and multiplied by the filter estimates of position and velocity to weight them appropriately. The probability-weighted estimates are added together to form blended estimates. This blending allows for partial failures in a sensor or combinations of failure types. A probability of one indicates that a filter is 100% accurate in its modeling and will completely determine the final blended estimates. A probability of zero indicates that a filter is completely inaccurate in its modeling and will not affect the blended estimates. The probability weighting computation block in Figure 1.5 represents the FDI block in Figure 1.3, and the multiple Kalman filters are in place of the single filter. MMAE is described in detail by Brown and Hwang (3) and is extensively used in several AFIT theses concerning stochastic estimation and control (7, 16, 17, 20, 28). An important aspect presented in these theses is the use of multiple model adaptive control (MMAC) for system stability and failure correction.

1.5.6 Literature Review Conclusion. The simple error detection presented in Section 1.5.1 would normally be effective for detecting failures, but the tracking errors in the code loop pseudorange caused by jamming or spoofing are small compared to other intrinsic

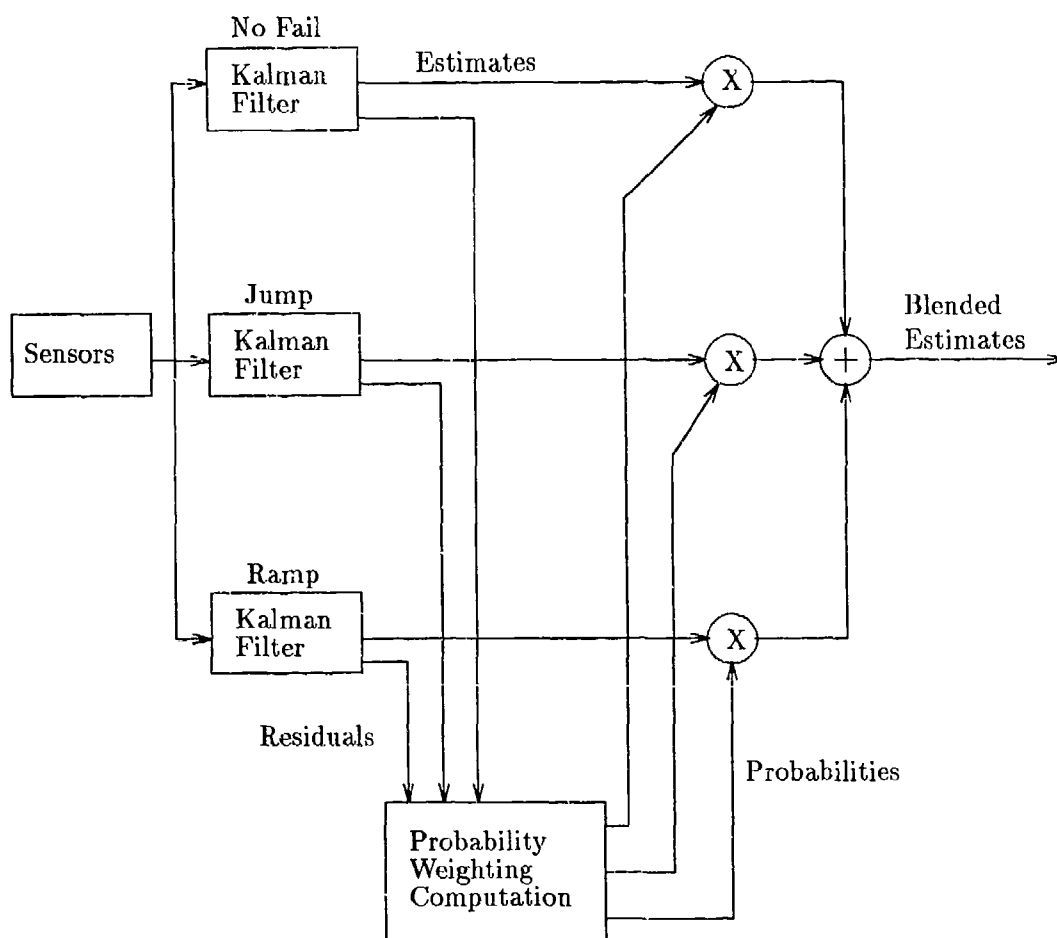


Figure 1.5. Multiple Model Adaptive Estimation

sic errors in the GPS. The performance of the GPS would be degraded, but the failures would not be detectable with this FDI technique (4).

Direct redundancy techniques are not feasible because multiple GPS receivers on a single aircraft are not practical. It would also be very expensive to support extra satellites in space designed to provide redundancy. The concept of analytic redundancy is inherent to the NRS design. The GPS receiver, INS measurement unit, transponders, and barometric altimeter are sensors which are mathematically related to generate multiple sources of the same information.

Multiple GLR testing and MMAE are prime candidates for the FDI system. Both of these techniques possess the versatility needed for the complex NRS, but the major

difference is the type of multiple filters required by each technique. Multiple GLR testing uses a bank of matching filters that are somewhat less complicated than the bank of Kalman filters used in MMAE. Both techniques were considered as final choices for the FDI system design and the GLR test was chosen. Additionally, a chi-square test is investigated to see if it can provide information in conjunction with the GLR algorithm. This leads to a multi-level FDI scheme in which the chi-square tests provides accurate and quick detection, while the GLR algorithm emphasizes isolation of the failure.

1.6 Methodology

The three main steps of the research approach are explained in Sections 1.6.1, 1.6.2 and 1.6.3. Steps one and two will satisfy the primary goal of the thesis, while step three is aimed at the follow-on goal included as a recommendation for future research.

1.6.1 Preliminary Studies. As mentioned in the literature review, multiple GLR testing has the advantage of needing only one Kalman filter, as compared to MMAE which requires a bank of such filters in parallel. Based on assumption 4, a simplified GLR test as formulated by Willsky (31) can be used under the stipulation that the failures must be additive in nature. This assumption may not be valid in computer simulated tests, or more importantly, in flight tests. Therefore, it is crucial that a preliminary study be performed on the GLR technique to verify its usefulness in this thesis.

A basic satellite orbit estimation problem can quickly test the GLR algorithm. The orbit problem is clearly defined by Maybeck (14:pp. 46-48) with range measured in radius units and time in time units. This study covers both ramp and jump failures in the range measurement at different magnitudes and rates, as shown in Table 1.1 for this example. The only goals of the example study were detection and identification of a failure in the range estimate of the satellite. No effort was made to adapt to any failures. The GLR algorithm performed favorably, so the multiple GLR testing concept was used rather than an MMAE configuration. No preliminary study was developed to test the MMAE technique.

Table 1.1. Orbit Runs

Run	Failure Type	Magnitude or Rate
1	None	N/A
2	Jump	0.03 radius units
3	Jump	0.5 radius units
4	Ramp	0.75 radius units/time unit
5	Ramp	1.0 radius units/time unit

1.6.2 System Tests. With the FDI technique chosen in the preliminary study, extensive system tests were performed on the NRS. The steps in these tests include:

1. Completely define the models for the GPS, INS, and RRS based on prior research by Negast, Snodgrass, Stacey (23, 26, 27).
2. Refine the FORTRAN code which represents these models and execute it with MSOFE.
3. Write computer code that will take the outputs of the MSOFE simulation and run the multiple GLR test algorithm.
4. Perform multiple simulations accounting for various failures and analyze the results.
5. Modify the design to get improved performance and return to step 2.

1.6.3 Adaptive Techniques. As mentioned, the follow-on goal is to determine corrective feedback based on the failure identification. This corrective feedback will improve the ability of the NRS to adapt to failures or uncertainties in the modeling of the NRS. A variety of techniques are available and some ideas are discussed in Chapter V even though time constraints prevented research into this area.

1.6.4 Stopping Criteria. There are three main criteria for termination that coincide with the three main steps in the research approach.

1. In order to prevent excessive time being spent on the preliminary study and jeopardizing the overall research effort, a suspense was placed on the pursuit of a working algorithm for GLR testing. If an adequate failure identification algorithm using

GLR testing did not work, then multiple GLR tests would be abandoned and MMAE would be pursued for the remainder of the thesis research. The GLR algorithm was successful in the preliminary studies and was pursued throughout the research.

2. With one of the FDI techniques fully developed, several factors will determine when the system tests are complete.

- (a) The design for the basic integrated NRS in a benign environment must provide accurate positions and velocities of the aircraft. Specifically, the NRS must determine the aircraft's latitude and longitude with a 1σ accuracy of 13 feet (ft) horizontal and 40 ft vertical. The 1σ accuracies for velocity are 0.1 ft/second (fps) in the north and east directions, and 0.4 fps vertically.

- (b) The FDI algorithm must be able to identify failures consistently. This criterion applies to all types of failures and variations in the magnitudes and rates of these failures. If this goal were not reached for certain failure types, then a determination would be made either to continue work on these failure types or to pursue adaptive techniques for the failures that have been accurately identified. The decision was made to focus on identification.

3. Adaptive techniques strive to meet two goals. First, stability of the NRS must be accomplished. Stability implies that the errors in the NRS estimates of position and velocity are not growing unbounded as time goes on. The second goal is to achieve the same accuracies listed in 2(a) above.

1.7 Overview of Thesis

Chapter II presents the detailed theory used in the research. Kalman filter theory is introduced with special attention on discretizing the dynamics of the sampled data Kalman filter. The basics of the FDI algorithms are discussed, including the equations implemented for the GLR and chi-square tests. Finally, the methods used for selecting thresholds are presented.

Chapter III fully describes the navigation system's parameters and operational details through an overall system description. Models for the NRS system to include the INS, GPS

and RRS are defined in detail. The failure models used in the simulations and the GLR algorithm are also introduced.

Results of the work done are shown in Chapter IV. The reduced order Kalman filter is analyzed, and a discussion of the FDI performance is presented. Chapter V summarizes the research through conclusions and recommendations.

II. Kalman Filtering and Failure Detection

2.1 Overview

This chapter presents the fundamental theory for application of a Kalman filter to the navigation problem. Basic filter equations will be presented for continuous-time dynamics followed by a discretization process to convert the filter for processing by the GLR algorithm. The equations for the GLR test will then show how the failure detection is accomplished. Threshold selection will be discussed along with tradeoffs in tuning the filter.

2.2 The Extended Kalman Filter

An extended Kalman filter (EKF) is chosen to provide state estimates depicting the dynamics of the NRS components. The EKF allows for nonlinear, time-varying dynamics and measurement vectors as found in this navigation problem. These vectors are linearized through approximation techniques about a nominal trajectory to form a linearized Kalman filter (LKF). The LKF is the basis for the EKF, which is found by linearizing about the updated state estimate rather than a nominal trajectory.

2.2.1 The Sampled Data Kalman Filter. Let the system model be expressed as a state equation of the form

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \quad (2.1)$$

where the state dynamics vector $\mathbf{f}[\mathbf{x}(t), t]$ is a nonlinear function of the state vector $\mathbf{x}(t)$ and time t . Let the process noise input matrix $\mathbf{G}(t) = \mathbf{I}$ and $\mathbf{w}(t)$ be a white Gaussian noise with mean:

$$\mathbf{m}_w = E\{\mathbf{w}(t)\} = \mathbf{0} \quad (2.2)$$

and noise strength $\mathbf{Q}(t)$ defined by:

$$E\{\mathbf{w}(t)\mathbf{w}^T(t + \tau)\} = \mathbf{Q}(t)\delta(\tau) \quad (2.3)$$

Incorporate measurements $\mathbf{z}(t_i)$ into the filter at discrete times and define them as a nonlinear function of the state vector and time:

$$\mathbf{z}(t_i) = \mathbf{h}[\mathbf{x}(t_i), t_i] + \mathbf{v}(t_i) \quad (2.4)$$

where $\mathbf{v}(t_i)$ is a zero-mean white Gaussian noise of covariance $\mathbf{R}(t_i)$ defined by:

$$E \{ \mathbf{v}(t_i) \mathbf{v}^T(t_j) \} = \begin{cases} \mathbf{R}(t_i) & \text{for } t_i = t_j \\ \mathbf{0} & \text{for } t_i \neq t_j \end{cases} \quad (2.5)$$

and $\mathbf{h}[\mathbf{x}(t_i), t_i]$ is the nonlinear observation vector. The LKF is based on *perturbation* states about a nominal state trajectory $\mathbf{x}_n(t)$ satisfying $\mathbf{x}_n(t_0) = \mathbf{x}_{n_0}$ and

$$\dot{\mathbf{x}}_n(t) = \mathbf{f}[\mathbf{x}_n(t), t] \quad (2.6)$$

where $\mathbf{f}[\mathbf{x}_n(t), t]$ is shown in Equation (2.1). The measurements are also based on the nominal states and defined as:

$$\mathbf{z}_n(t_i) = \mathbf{h}[\mathbf{x}_n(t_i), t_i] \quad (2.7)$$

The perturbation states are found by subtracting the nominal states in Equation (2.6) from the original states in Equation (2.1):

$$[\dot{\mathbf{x}}(t) - \dot{\mathbf{x}}_n(t)] = \mathbf{f}[\mathbf{x}(t), t] - \mathbf{f}[\mathbf{x}_n(t), t] + \mathbf{G}(t)\mathbf{w}(t) \quad (2.8)$$

Equation (2.8) is approximated to first order through a truncated Taylor series expansion:

$$\dot{\delta \mathbf{x}}(t) = \mathbf{F}[t; \mathbf{x}(t)] \delta \mathbf{x}_n(t) + \mathbf{G}(t)\mathbf{w}(t) \quad (2.9)$$

where $\delta \mathbf{x}(t)$ are the perturbation states. The definitions for $\mathbf{G}(t)$ and $\mathbf{w}(t)$ are unchanged and the new linearized dynamics matrix $\mathbf{F}[t; \mathbf{x}_n(t)]$ is found by taking partial derivatives of $\mathbf{f}[\mathbf{x}(t), t]$ with respect to $\mathbf{x}(t)$ and evaluated at the nominal values for the trajectory $\mathbf{x}_n(t)$:

$$\mathbf{F}[t; \mathbf{x}_n(t)] = \left. \frac{\partial \mathbf{f}[\mathbf{x}(t), t]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_n(t)} \quad (2.10)$$

The discrete-time measurements are similarly approximated to first order and in the perturbed form:

$$\delta \mathbf{z}(t_i) = \mathbf{H}[t_i; \mathbf{x}(t_i)] \delta \mathbf{x}_n(t) + \mathbf{v}(t_i) \quad (2.11)$$

and the same linearization process is used for the measurement matrix $\mathbf{H}[t_i; \mathbf{x}_n(t_i)]$, resulting in:

$$\mathbf{H}[t_i; \mathbf{x}_n(t_i)] = \left. \frac{\partial \mathbf{h}[\mathbf{x}(t_i), t_i]}{\partial \mathbf{x}} \right|_{\mathbf{x} = \mathbf{x}_n(t_i)} \quad (2.12)$$

The LKF in this thesis generates error state estimates $\widehat{\delta \mathbf{x}}(t)$ which can be added to the nominal states to provide whole states estimates $\hat{\mathbf{x}}(t)$ in the form:

$$\hat{\mathbf{x}}(t) = \mathbf{x}_n(t) + \widehat{\delta \mathbf{x}}(t) \quad (2.13)$$

The EKF will now be formed by linearizing about the state estimate $\hat{\mathbf{x}}$ rather than the nominal trajectory \mathbf{x}_n . The following equations use the notation t/t_i to represent the time history of a given variable conditioned on the measurements taken through the time interval $[t_i, t_{i+1})$. Also, t_i^- represents the value after propagation but prior to the measurement update and t_i^+ corresponds to the value after the measurement update. The state estimates and covariance values $\mathbf{P}(t/t_i)$ are propagated from t_i to t_{i+1} by solving the following differential equations:

$$\dot{\hat{\mathbf{x}}}(t/t_i) = \mathbf{f}[\hat{\mathbf{x}}(t/t_i), t] \quad (2.14)$$

$$\dot{\mathbf{P}}(t/t_i) = \mathbf{F}[t; \hat{\mathbf{x}}(t/t_i)]\mathbf{P}(t/t_i) + \mathbf{P}(t/t_i)\mathbf{F}^T[t; \hat{\mathbf{x}}(t/t_i)] + \mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^T(t) \quad (2.15)$$

where

$$\mathbf{F}[t; \hat{\mathbf{x}}(t/t_i)] = \left. \frac{\partial \mathbf{f}[\mathbf{x}(t), t]}{\partial \mathbf{x}} \right|_{\mathbf{x} = \hat{\mathbf{x}}(t/t_i)} \quad (2.16)$$

and initial conditions are given by:

$$\hat{\mathbf{x}}(t_i/t_i) = \hat{\mathbf{x}}(t_i^+) \quad (2.17)$$

$$\mathbf{P}(t_i/t_i) = \mathbf{P}(t_i^+) \quad (2.18)$$

The discrete-time measurements are processed in the EKF through update equations:

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-) \mathbf{H}^T[t_i; \hat{\mathbf{x}}(t_i^-)] \left\{ \mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)] \mathbf{P}(t_i^-) \mathbf{H}^T[t_i; \hat{\mathbf{x}}(t_i^-)] + \mathbf{R}(t_i) \right\}^{-1} \quad (2.19)$$

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_i) \{ \mathbf{z}_i - \mathbf{h}[\hat{\mathbf{x}}(t_i^-), t_i] \} \quad (2.20)$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i) \mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)] \mathbf{P}(t_i^-) \quad (2.21)$$

where

$$\mathbf{H}(t_i) = \mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)] = \left. \frac{\partial \mathbf{h}[\mathbf{x}(t_i), t_i]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}(t_i^-)} \quad (2.22)$$

and $\mathbf{K}(t_i)$ is the discrete-time Kalman filter gain. Note that, for the EKF, the measurement and dynamics vectors are calculated about the last state estimate $\hat{\mathbf{x}}(t_i^-)$ rather than the nominal trajectory used by a simple linearized Kalman filter.

2.2.2 The Discrete-Time Kalman Filter. In order to utilize the filter outputs in the GLR algorithm, it is necessary to discretize the state dynamics matrix into a state transition matrix (STM), $\Phi(t_i, t_{i-1})$. All other quantities of interest such as \mathbf{K} and \mathbf{H} are already in discrete form. The STM must satisfy the following differential equation and initial condition (13):

$$d[\Phi(t, t_{i-1})]/dt = \mathbf{F}(t) \Phi(t, t_{i-1}) \quad (2.23)$$

$$\Phi(t_{i-1}, t_{i-1}) = \mathbf{I} \quad (2.24)$$

Defining $\Delta t = t_i - t_{i-1}$ and solving with \mathbf{F} assumed constant over Δt (see assumption 6) leads to:

$$\Phi(t_i, t_{i-1}) = e^{\mathbf{F} \Delta t} \quad (2.25)$$

The state equation can now be written in the discrete form

$$\delta \mathbf{x}(t_i) = \Phi(t_i, t_{i-1}) \delta \mathbf{x}(t_{i-1}) + \mathbf{G}_d(t_{i-1}) \mathbf{w}_d(t_{i-1}) \quad (2.26)$$

where \mathbf{G}_d and \mathbf{w}_d are discrete-time representations of \mathbf{G} and \mathbf{w} defined earlier.

2.3 Failure Detection

This section will present the theory behind GLR and chi-square testing for the purpose of failure detection. Given the Kalman filter developed in Sections 2.1 and 2.2, an algorithm can be used to observe changes in the residuals. If the changes are significant, they will represent failures in the system by causing the GLR or chi-square value to exceed a threshold. Windowing will be applied as discussed in Chapter I.

2.3.1 GLR Equations. The primary goal is to define a likelihood function $l(t_i, \theta)$ that, when compared to a threshold, will identify the onset of a failure such as jamming or spoofing. Two hypotheses are established with a Kalman filter based on H_0 (no failure) and matching filters based on H_1 (a failure has occurred). The Kalman filter state equation from Section 2.2 is in the discrete form

$$\delta \mathbf{x}^0(t_i) = \Phi(t_i, t_{i-1}) \delta \mathbf{x}^0(t_{i-1}) + \mathbf{G}_d(t_{i-1}) \mathbf{w}_d(t_{i-1}) \quad (2.27)$$

with discrete measurements described by

$$\delta \mathbf{z}^0(t_i) = \mathbf{H}(t_i) \delta \mathbf{x}^0(t_i) + \mathbf{v}(t_i) \quad (2.28)$$

The matching filters will not provide state estimation but are designed for failure detection and will have the form

$$\delta \mathbf{x}^1(t_i) = \Phi(t_i, t_{i-1}) \delta \mathbf{x}^1(t_{i-1}) + \mathbf{G}_d(t_{i-1}) \mathbf{w}_d(t_{i-1}) \quad (2.29)$$

and

$$\delta \mathbf{z}^1(t_i) = \mathbf{H}(t_i) \delta \mathbf{x}^1(t_i) + \mathbf{v}(t_i) + \mathbf{d}(t_i) n(t_i, \theta) \nu \quad (2.30)$$

where

$\mathbf{d}(t_i)$	=	failure vector
$n(t_i, \theta)$	=	failure function
ν	=	unknown size of the failure
θ	=	unknown time of the failure

Comparison of Equations (2.28) and (2.30) shows that a matching filter can assume a failure in the system by modeling it as some variation in the actual measurement beyond the variations caused by dynamics of the system. Although the failure is modeled as a bias on the measurement, this model can also represent changes in the states caused by real world anomalies. Further definition of the new failure quantities in Equation (2.30) will show how the failures are modeled in the matching filter. The failure vector $\mathbf{d}(t_i)$ is r -by-1 where r is the number of measurements. The 1's in the failure vector indicate which measurement devices are assumed to be induced with a failure and the other elements of $\mathbf{d}(t_i)$ are zero. The failure function $n(t_i, \theta)$ tells the matching filter where the failure is assumed to occur within the window (see assumption 12) and the form of the failure such as a step function. This allows the generation of a different GLR based on different failure times. For example, if the failure is assumed to be a unit step and to occur 3 time units from the front of the window, then $\theta = 3$ and

$$\begin{aligned} n(t_i, \theta) &= 1 \quad \text{for } \theta \geq 3 \\ &= 0 \quad \text{for } \theta < 3 \end{aligned} \tag{2.31}$$

Finally, the size ν simply dictates the unknown magnitude of the failure, whereas $n(t_i, \theta)$ and $\mathbf{d}(t_i)$ are predetermined design parameters. It is important to note that, by not making ν a predetermined constant, it will actually be estimated by the GLR algorithm as shown later and can be used to provide corrective feedback. Section 3.4 provides a detailed discussion of how the failures are modeled using these equations.

In general, the likelihood function or GLR, $l(t_i, \theta)$, is based on maximum likelihood estimates of θ and ν designated as $\hat{\theta}$ and $\hat{\nu}$. When considering all possible values of θ within the window, the GLR with the largest value indicates the presence *and* time of the failure. The derivation shown by Riggins (25:pp. 112-115) illustrates that in forming $l(t_i, \theta)$, it is inherently maximized over ν but obtaining an MLE of θ is based on the definition of $n(t_i, \theta)$. With $n(t_i, \theta) = 1$ for all t_i , only a single likelihood function will be calculated and it will not detect the failure until it reaches the beginning of the window. Therefore,

the GLR's are based on $\hat{\nu}$ but not $\hat{\theta}$. The primary reason for this decision is to reduce the computation time that increases significantly when calculating several GLR's based on different values of θ .

The Kalman filter residuals $\gamma(t_i)$ are shown in Equation (2.20) and are defined by

$$\gamma(t_i) = \mathbf{z}_i - \mathbf{h}[\hat{\mathbf{x}}(t_i^-), t_i] \quad (2.32)$$

and these residuals can be expressed for each hypothesis as

$$H_0 : \gamma(t_i) = \gamma^0(t_i) \quad H_1 : \gamma(t_i) = \gamma^0(t_i) + \mathbf{m}(t_i, \theta)\nu \quad (2.33)$$

For a Kalman filter successfully tracking the true states, $\gamma^0(t_i)$ will appear as zero-mean white Gaussian noise of covariance $\mathbf{\Lambda} = \mathbf{H}\mathbf{P}^-\mathbf{H}^T + \mathbf{R}$. With a failure induced on the measurements, a signal of unknown magnitude, $\mathbf{m}(t_i, \theta)\nu$, will also be present in the residuals with ν defined earlier and $\mathbf{m}(t_i, \theta)$ presented momentarily. It is the goal of the GLR algorithm to identify this signal by recognizing variations in the residuals from their normal unfailed values. The GLR tests are particularly good at detecting jumps in the residuals with the key being how closely the matching filters model the actual failures. Section 1.5.4 stated that the GLR algorithm is a function of overall system behavior (Φ and \mathbf{H}) and Kalman filter gain \mathbf{K} . This is shown mathematically in Equations (2.34) and (2.35) with the derivation shown by Riggins (25:pp. 112-115). The failure residual offset $\mathbf{m}(t_i, \theta)$ is found through

$$\mathbf{m}(t_i, \theta) = \mathbf{H}(t_i)\mathbf{y}(t_i, \theta) + \mathbf{d}(t_i)n(t_i, \theta) \quad (2.34)$$

where the recursive failure quantity $\mathbf{y}(t_{i+1}, \theta)$ is given by

$$\mathbf{y}(t_{i+1}, \theta) = \Phi(t_i, t_{i-1})[\mathbf{I} - \mathbf{K}(t_i)\mathbf{H}(t_i)]\mathbf{y}(t_i, \theta) - \Phi(t_i, t_{i-1})\mathbf{K}(t_i)\mathbf{d}(t_i)n(t_i, \theta) \quad (2.35)$$

With the failure assumed to occur at the beginning of the window (see assumption 12), Equations (2.34) and (2.35) can be simplified by setting $n(t_i, \theta)=\text{one}$ for all t_i

$$\mathbf{y}(t_{i+1}, \theta) = \Phi(t_i, t_{i-1})[\mathbf{I} - \mathbf{K}(t_i)\mathbf{H}(t_i)]\mathbf{y}(t_i, \theta) - \Phi(t_i, t_{i-1})\mathbf{K}(t_i)\mathbf{d}(t_i) \quad (2.36)$$

$$\mathbf{m}(t_i) = \mathbf{H}(t_i)\mathbf{y}(t_i) + \mathbf{d}(t_i) \quad (2.37)$$

The consequence of this simplification is a delay in detecting a failure because the failure is not realized until it reaches the beginning of the window. The combination of the Kalman filter outputs and the matching filter model will determine the magnitude of the likelihood function defined as:

$$l(t_i, \theta) = S^T(t_i, \theta)C^{-1}(t_i, \theta)S(t_i, \theta) \quad (2.38)$$

where

$$S(t_i, \theta) = \sum_{j=1}^i \mathbf{m}^T(t_j, \theta)\mathbf{\Lambda}^{-1}(t_j)\gamma(t_j) \quad (2.39)$$

$$C(t_i, \theta) = \sum_{j=1}^i \mathbf{m}^T(t_j, \theta)\mathbf{\Lambda}^{-1}(t_j)\mathbf{m}(t_j, \theta) \quad (2.40)$$

given

$$\mathbf{\Lambda}(t_j) = \mathbf{H}(t_j)\mathbf{P}(t_j^-)\mathbf{H}^T(t_j) + \mathbf{R} \quad (2.41)$$

and the MLE of the unknown magnitude of the failure, ν , is found by:

$$\hat{\nu}(t_i, \theta) = \frac{S(t_i, \theta)}{C(t_i, \theta)} \quad (2.42)$$

The residual covariance $\mathbf{\Lambda}(t_j)$ and the residuals are combined with Equations (2.34) and (2.35) or Equations (2.36) and (2.37) to give a linear combination of the residuals $S(t_i, \theta)$ and a deterministic value $C(t_i, \theta)$ defined in Equations (2.39) and (2.40). Finally, a decision rule based on a threshold, ϵ , would be

$$\begin{aligned} l(t_i, \theta) &> \epsilon \Rightarrow \text{FAILURE} \\ l(t_i, \theta) &\leq \epsilon \Rightarrow \text{NO FAILURE} \end{aligned} \quad (2.43)$$

2.3.2 Chi-Square Equations. A chi-square test will be used and is based on the Kalman filter residuals $\gamma(t_j)$ which are zero mean and white with known residual covariance

$\Lambda(t_j)$. The chi-square random variable $\chi(t_k)$ is given by

$$\chi(t_k) = \sum_{j=k-N+1}^k \gamma^T(t_j) \Lambda^{-1}(t_j) \gamma(t_j) \quad (2.44)$$

with N being the size of a sliding window. Notice that the system dynamics are not included in Equation (2.44) and that only one failure hypothesis is available. This agrees with the discussion in Section 1.5.4 about the simplicity of the chi-square algorithm. A detection rule based on an established threshold ϵ would be

$$\begin{aligned} \chi(t_k) &> \epsilon \Rightarrow \text{FAILURE} \\ \chi(t_k) &\leq \epsilon \Rightarrow \text{NO FAILURE} \end{aligned} \quad (2.45)$$

2.3.3 Threshold Selection and Filter Tuning. All thresholds used in the FDI logic will be determined empirically. Three major concerns will be considered in selecting final threshold values. First, the thresholds must be low enough to prevent delays in detecting failures. Similarly, they must be low enough to prevent missed alarms caused by GLR or χ values dropping below the threshold while an actual failure is still present. Finally, the thresholds must be high enough to prevent false alarms caused by variations in the GLR and χ values. These variations may result from aircraft maneuvers or unpredictable changes in the random measurement noise. If the filter is tuned sufficiently for both good tracking and enhanced failure detection, then these variations or *noise floor* will be relatively low.

When tuning the Kalman filter, a major tradeoff must be made to meet the goals for state estimation and failure detection. Adjusting the process noise and measurement noise values to enhance FDI capabilities may degrade state estimation and vice versa. This can be seen in Equation (2.41) by realizing that reductions in \mathbf{R} will cause reductions in Λ , resulting in better possible monitoring of the residuals. The consequence is that reductions in \mathbf{R} will cause an increase in $\frac{Q}{R}$, resulting in a conservatively tuned filter. The primary goal of the NRS filter is to provide a navigation solution within the operational specifications listed in Chapter I. Once this goal is met, the measurement noise for the filter can be reduced to allow better FDI, while the process noise is increased to maintain good

tuning of the filter. Observation of the state estimates, measurement residuals and the residual covariance $\Lambda(t_i)$ will indicate a point of diminishing returns using this technique. Once the filter has been tuned via multi-run simulations to provide optimum residual monitoring without seriously degrading state estimation, single-run simulations will be used to evaluate actual FDI performance.

2.4 Summary

This chapter provided the theoretical basis for the remaining chapters. The actual models for the Kalman filter and the system are presented in Chapter III along with the details of the matching filter design for the GLR algorithm. Verification of the discretization process is shown in Chapter IV. Also included in Chapter IV, are the results obtained by applying the equations discussed in Chapter II, including the filter performance, FDI performance and tradeoffs encountered concerning threshold selection and filter tuning.

III. Navigation and Failure Models

3.1 Overview

This chapter describes the models for each of the three navigation systems and the failure models. An overall system description is given, followed by detailed state and measurement equations. Finally, the methods used to simulate the various failures are shown as they apply to the theory in Chapter II.

3.2 Overall System Description

A brief reiteration of the basic elements in the system is helpful for this discussion. The three navigation systems are the GPS, INS, and RRS. There are 14 measurements provided to the Kalman filter, including four satellite vehicle (SV) pseudoranges, six transponder ranges, three-axis velocity aiding from a Doppler system and altitude from the barometric altimeter. A total of 97 error states about nominal values make up the *truth model* to represent the real world. A total of 15 error states are used for the Kalman filter model.

A block diagram depicting the NRS truth and filter models is shown in Figure 3.1. The *true* aircraft position \mathbf{x} is generated by PROFGEN and provided to each navigation system. The SV positions are generated by ORBIT and combined with the true aircraft position to obtain *pseudoranges* for use by the GPS. Each navigation system generates perturbations from the true range and the final *difference* measurements are then formed by subtracting the GPS and RRS measured ranges from their corresponding INS calculated ranges. The EKF propagates equations that represent the NRS and uses the measurements to update estimates of the error states. Finally, these estimates are used to correct the original INS indicated position.

3.3 Model Descriptions

The truth model consists of 41 INS states, 26 RRS states, and 30 GPS states. The filter model consists of 11 INS states, two RRS states, and two GPS states. The following sections will provide details and justification for these model selections.

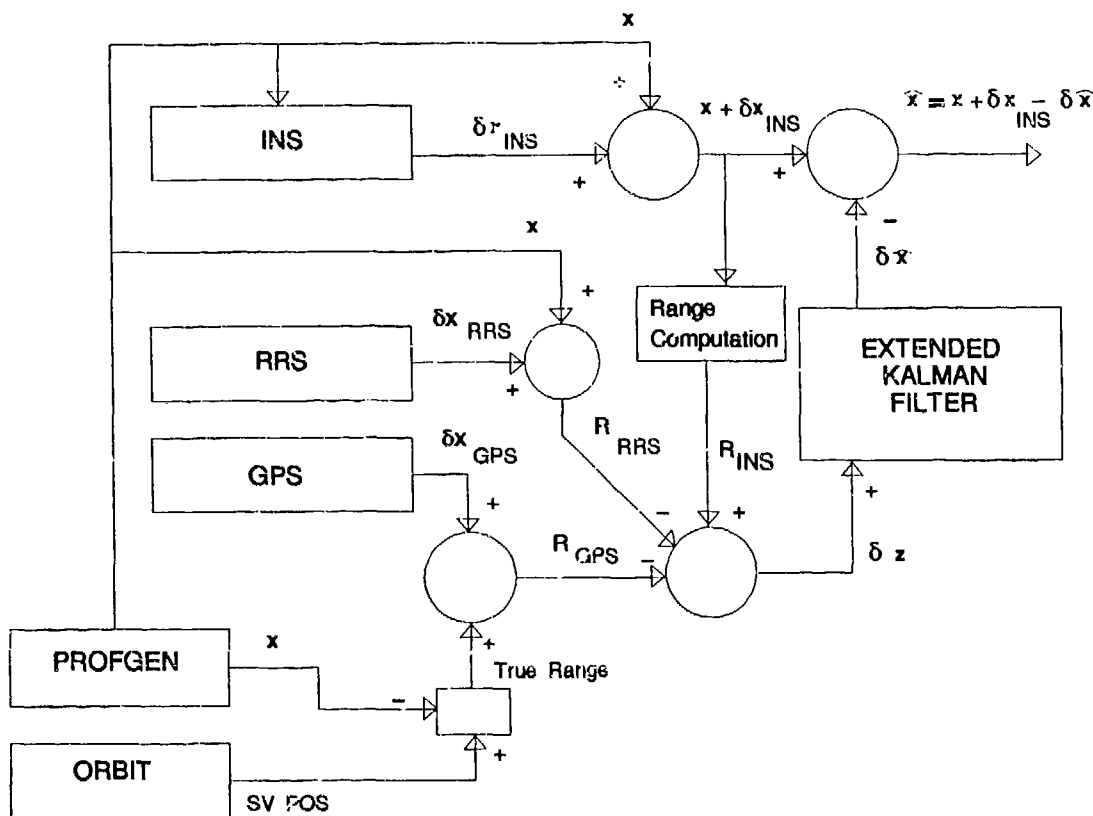


Figure 3.1. Truth and Filter Model Block Diagram

3.3.1 INS Model. This section presents the truth and filter models used for the INS. The INS is a strapped-down wander azimuth system that senses aircraft motion via gyros and accelerometers and is used as the primary source for navigation. A 93-state model is presented with specific interest in the 41 states kept for the truth model. The reduced order filter model is then discussed.

3.3.1.1 The 93-State LN-93 Error Model. The 41 INS states were extracted from an original 93-State INS model based on the Litton LN-93 error state model which is described using six categories of states:

$$\delta x = [\delta x_1^T \quad \delta x_2^T \quad \delta x_3^T \quad \delta x_4^T \quad \delta x_5^T \quad \delta x_6^T]^T \quad (3.1)$$

where $\delta \mathbf{x}$ is a 93×1 column vector and:

$\delta \mathbf{x}_1$ represents the "general" error vector containing 13 position, velocity, attitude, and vertical channel errors.

$\delta \mathbf{x}_2$ consists of 16 gyro, accelerometer, and baro-altimeter exponentially time-correlated errors, and "trend" states. These states are modeled as first order Markov processes in the truth (system) model.

$\delta \mathbf{x}_3$ represents gyro bias errors. These 18 states are modeled as random constants in the truth model.

$\delta \mathbf{x}_4$ is composed of the accelerometer bias error states. These 22 states are modeled in exactly the same manner as the gyro bias states.

$\delta \mathbf{x}_5$ depicts accelerometer and gyro initial thermal transients. The 6 thermal transient states are first order Markov processes in the system model.

$\delta \mathbf{x}_6$ models the gyro compliance errors. These 18 error states are modeled as biases in the system model.

The truth model system state space differential equation is given by:

$$\begin{Bmatrix} \delta \dot{\mathbf{x}}_1 \\ \delta \dot{\mathbf{x}}_2 \\ \delta \dot{\mathbf{x}}_3 \\ \delta \dot{\mathbf{x}}_4 \\ \delta \dot{\mathbf{x}}_5 \\ \delta \dot{\mathbf{x}}_6 \end{Bmatrix} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & \mathbf{F}_{13} & \mathbf{F}_{14} & \mathbf{F}_{15} & \mathbf{F}_{16} \\ \mathbf{0} & \mathbf{F}_{22} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F}_{55} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \delta \mathbf{x}_1 \\ \delta \mathbf{x}_2 \\ \delta \mathbf{x}_3 \\ \delta \mathbf{x}_4 \\ \delta \mathbf{x}_5 \\ \delta \mathbf{x}_6 \end{Bmatrix} + \begin{Bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \quad (3.2)$$

A full description of the submatrices for this equation is given in the Litton LN-93 manual (9).

3.3.1.2 The 41-State INS Error Model. The large number of states in the LN-93 model results in a truth model that is cumbersome to the simulation software. Studies by Negast and others have shown that 41 states are sufficient to represent the INS

truth model accurately (10, 23). With only 41 of the original 93 states selected for use in this thesis, the submatrices in Equation (3.2) are reduced significantly. Appendix A, Tables A.1 and A.2 show the 41 states chosen for this thesis and Appendix B presents the equations and state dynamics noise values used in the submatrices.

3.3.1.3 INS Measurement Models The two measurements associated with the INS are the barometric altimeter aiding and the Doppler based velocity aiding. As mentioned previously, the altimeter aiding is used to overcome the instability inherent in the vertical channel of the INS. The altimeter output is modeled as the sum of the true altitude h_t , the total error in the barometric altimeter δh_B , and a random measurement noise v . Similarly, the INS calculated altitude is the sum of the true altitude and the INS error in vehicle altitude above the reference ellipsoid, δh . A *difference* measurement is used to eliminate the unknown true altitude, h_t , resulting in Equation 3.3.

$$\begin{aligned}\delta z &= [h_t + \delta h] - [h_t + \delta h_B] + v \\ &= [1]\delta h - [1]\delta h_B + v\end{aligned}\tag{3.3}$$

A perfect Doppler system provides velocity aiding to the INS based on assumption 13. The Doppler aiding could come from the GPS or a simple radar system. This measurement source is not particularly significant for most of the thesis, but it does allow the filter to generate better estimates of the velocity states. A simple model is assumed for the Doppler measurement. All three channels (north, east, and up) are represented by the difference between the *truth* state velocity error, δV_{ti} , and the *filter* state velocity error, δV_i , as shown in Equation (3.4).

$$\delta z_i = \delta V_{ti} - \delta V_i \quad \text{where } i = x, y, z\tag{3.4}$$

Although this model seems somewhat unrealistic in that it provides the filter with an ideal measurement for velocities, it does not skew the performance of the FDI algorithm because these measurements are not used in the FDI calculations. The primary reason for including the Doppler measurements was for comparison of the filter performance against prior AFIT theses (23, 26, 27).

3.3.1.4 *The Reduced 11-State INS Filter Model.* The number of states in the Kalman filter is further reduced from the 41-state truth model. Consideration was given to magnitudes of the states and their estimability when deciding which states could be eliminated for the filter. The reduced order filter had to be tuned to compensate for the eliminated states by adjusting dynamics noise and measurement noise values. Table A.5 shows the 11-states used for the INS filter model and Tables B.10 and B.11 show the final tuning parameters used in the filter.

3.3.2 *The 26-State RRS Error Model.* The Range/Range-Rate System (RRS) navigation aiding system is modeled using 26 states for six ground-based transponders (23, 27). The RRS *interrogates* the transponders collecting the electromagnetic (EM) signals they emit. These signals represent position information used to calculate a navigation solution which supports the INS. Table A.3 shows the 26 states and their corresponding state number in the overall 97-state truth model. There are two states common to each transponder and four which are modeled separately for each transponder.

3.3.2.1 *RRS Model Equations.* The two common states for the transponders are a result of errors in user hardware. They appear as bias terms and are modeled as random constants. Their state equations are given by:

$$\begin{Bmatrix} \dot{x}_{br} \\ \dot{x}_{bv} \end{Bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} x_{br} \\ x_{bv} \end{Bmatrix} \quad (3.5)$$

where

x_{br}	=	range equivalent of interrogator bias
x_{bv}	=	velocity equivalent of interrogator bias

The initial conditions for these states were chosen to be consistent with previous AFIT research (23, 26, 27) and are:

$$\begin{Bmatrix} \hat{x}_{br}(t_0) \\ \hat{x}_{bv}(t_0) \end{Bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (3.6)$$

and

$$\mathbf{P}_{br,bv}(t_0) = \begin{bmatrix} 1ft^2 & 0 \\ 0 & 10^{-4}ft^2/sec^2 \end{bmatrix} \quad (3.7)$$

The four states which are unique to each transponder represent the error in the surveyed position (x,y, and z) of the transponders and the atmospheric propagation delay between the transponder and the receiving aircraft or user. The position errors are modeled as random constants and the atmospheric error is represented by a first order Markov process. The state equations for these error sources are shown below with i used to represent the various transponders:

$$\begin{Bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \\ \dot{\delta R}_{atm_i} \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{300sec} \end{bmatrix} \begin{Bmatrix} x_i \\ y_i \\ z_i \\ \delta R_{atm_i} \end{Bmatrix} + \begin{Bmatrix} w_{x_i} \\ w_{y_i} \\ w_{z_i} \\ w_{atm_i} \end{Bmatrix} \quad (3.8)$$

The initial conditions for these states were chosen to be consistent with previous AFIT research (23, 26, 27) and are:

$$\hat{\mathbf{x}}_{x,y,z,atm}(t_0) = \mathbf{0} \quad (3.9)$$

$$\mathbf{P}_{x,y,z,atm}(t_0) = \begin{bmatrix} 25ft^2 & 0 & 0 & 0 \\ 0 & 25ft^2 & 0 & 0 \\ 0 & 0 & 25ft^2 & 0 \\ 0 & 0 & 0 & 100(PPM)^2 \end{bmatrix} \quad (3.10)$$

and

$$E \{ \mathbf{w}_{x,y,z,atm}(t) \} = \mathbf{0} \quad (3.11)$$

$$E \{ \mathbf{w}_{x,y,z,atm}(t) \mathbf{w}_{x,y,z,atm}^T(t + \tau) \} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{2\sigma_{atm}^2}{300} \end{bmatrix} \delta(\tau) \quad (3.12)$$

with $\sigma_{atm}^2 = 10^{-10} \text{ ft}^2$.

3.3.2.2 RRS Measurement Model. The system description identified measurement sources which included RRS ranges. The RRS measurements indicate the range from the transponders to the user and Figure 3.2 shows the errors in the *true* positions. This measurement is expressed in Equation (3.13) as the sum of the true range, error sources, and a random measurement noise v .

$$R_{RRS} = R_t + \delta R_{atm} + \delta R_b + v \quad (3.13)$$

where

R_{RRS}	=	RRS range measurement, from transponder to user
R_t	=	true range, from transponder to user
δR_{atm}	=	range error due to atmospheric delay
δR_b	=	error due to equipment bias
v	=	zero-mean white Gaussian measurement noise

The true range R_t is not actually known, so a *difference* measurement, δz must be obtained using the range, R_{INS} , from the transponder to the user. This range is not a measurement, but is calculated by the INS using Equation (3.14).

$$R_{INS} = \left| \mathbf{X}_U - \mathbf{X}_T \right| = \left| \begin{bmatrix} x_U \\ y_U \\ z_U \end{bmatrix} - \begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} \right| \quad (3.14)$$

where \mathbf{X}_U and \mathbf{X}_T represent the user and transponder position vectors in the earth centered earth fixed (ECEF) frame respectively. Another way to write Equation (3.14) is:

$$R_{INS} = \sqrt{(x_U - x_T)^2 + (y_U - y_T)^2 + (z_U - z_T)^2} \quad (3.15)$$

Figure 3.2 shows that \mathbf{X}_U and \mathbf{X}_T are not completely known and have some error. Based on assumption 11 with perturbations representing the errors in \mathbf{X}_U and \mathbf{X}_T , Equation (3.15) can be approximated and written in terms of the true range and a truncated first-order Talyor series:

$$\begin{aligned} R_{INS} = R_t &+ \left. \frac{\partial R_{INS}(\mathbf{X}_T, \mathbf{X}_U)}{\partial \mathbf{X}_T} \right|_{(\mathbf{X}_T, \mathbf{X}_U)_{nom}} \cdot \delta \mathbf{X}_T \\ &+ \left. \frac{\partial R_{INS}(\mathbf{X}_T, \mathbf{X}_U)}{\partial \mathbf{X}_U} \right|_{(\mathbf{X}_T, \mathbf{X}_U)_{nom}} \cdot \delta \mathbf{X}_U \end{aligned} \quad (3.16)$$

The solution for R_{INS} is found by substituting Equation (3.15) into Equation (3.16) and evaluating the partial derivatives to get:

$$\begin{aligned} R_{INS} = R_t &- \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_U - \left[\frac{y_T - y_U}{|R_{INS}|} \right] \cdot \delta y_U - \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_U \\ &+ \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_T + \left[\frac{y_T - y_U}{|R_{INS}|} \right] \cdot \delta y_T + \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_T \end{aligned} \quad (3.17)$$

Finally, the *difference* measurement is given as:

$$\begin{aligned} \delta z &= R_{INS} - R_{RRS} \\ &= - \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_U - \left[\frac{y_T - y_U}{|R_{INS}|} \right] \cdot \delta y_U - \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_U \\ &+ \left[\frac{x_T - x_U}{|R_{INS}|} \right] \cdot \delta x_T + \left[\frac{y_T - y_U}{|R_{INS}|} \right] \cdot \delta y_T + \left[\frac{z_T - z_U}{|R_{INS}|} \right] \cdot \delta z_T \\ &- [1] \delta R_{atm} - [1] \delta R_b - v \end{aligned} \quad (3.18)$$

3.3.2.3 The Reduced Two-State RRS Filter Model. The same goal of reducing the number of states in the filter was met with the RRS model. Research at CIGTF has

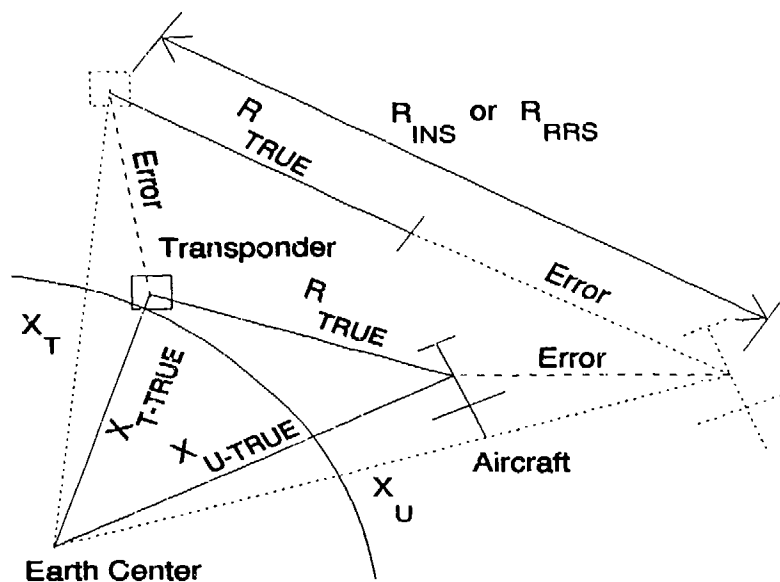


Figure 3.2. RRS Measurements

shown that retaining only the first two states, which are common to all the transponders, can provide good filter performance. These states are the range and velocity errors due to equipment bias and are represented as δR_b in Equation (3.18). Previous AFIT researchers have kept all 26 states in the filter because state reduction was not a major goal. The scenario for this thesis requires a filter of few states, so an effort was made to prove that the two-state filter was adequate for navigation through comparison to the 26-state model results obtained by Negast (23). Filter tuning included increases in strengths of dynamics and measurement noises with final values shown in Tables B.10 and B.11.

3.3.3 The 30-State GPS Error Model. The third and final navigation system is based on EM signals transmitted from orbiting satellites. Although similar in concept to the RRS, the GPS is modeled somewhat differently. This model has been developed

through research at AFIT and many of its fundamental concepts are addressed in a variety of sources (12, 23, 26, 27). The dynamics and measurement equations for the full 30-state truth model are presented followed by the reduced two-state filter model. A tabular listing of the 30-state model is shown in Table A.4 and the two filter states chosen are listed in Table A.5.

3.3.3.1 GPS Model Equations. There are five types of error sources that are modeled. The first two states represent the errors in the user clock and are modeled as follows:

$$\begin{Bmatrix} \dot{x}_{Uclk_b} \\ \dot{x}_{Uclk_{dr}} \end{Bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} x_{Uclk_b} \\ x_{Uclk_{dr}} \end{Bmatrix} \quad (3.19)$$

where

$$\begin{aligned} x_{Uclk_b} &= \text{range equivalent of user set clock bias} \\ x_{Uclk_{dr}} &= \text{velocity equivalent of user set clock drift} \end{aligned}$$

The initial state estimates and covariances for these states were chosen to be consistent with previous AFIT research (23, 26, 27) and are:

$$\begin{Bmatrix} \hat{x}_{Uclk_b}(t_0) \\ \hat{x}_{Uclk_{dr}}(t_0) \end{Bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (3.20)$$

and

$$\mathbf{P}_{Uclk_b, Uclk_{dr}}(t_0) = \begin{bmatrix} 9.0 \times 10^{14} ft^2 & 0 \\ 0 & 9.0 \times 10^{10} ft^2/sec^2 \end{bmatrix} \quad (3.21)$$

Because these error sources are a function of the user equipment, they are common to all the SV's. The remaining four types of errors are unique to each SV, based on their individual equipment and their position with respect to the user. One SV specific error source is the code loop error. Although the code loop is part of the user equipment shared by all the SV's, its error magnitude is relative to each SV. Next is the atmospheric interference with the EM signals as related to the ionospheric and tropospheric delay in

the signals propagation. The code loop error, tropospheric delay, and ionospheric delay are all modeled as first order Markov processes with time constants shown in Equation (3.22). All three are driven by zero-mean white Gaussian noise with strengths shown in Equation (3.25).

The fourth error source is due to inaccuracies in the clocks on board the SV's and the final error source is based on line-of-sight errors between the SV's and the receiver. The models for these states are shown in Equation (3.22) - (3.25).

$$\begin{pmatrix} \dot{\delta R}_{cl} \\ \dot{\delta R}_{trop} \\ \dot{\delta R}_{ion} \\ \dot{\delta R}_{selk} \\ \dot{\delta x}_{s_i} \\ \dot{\delta y}_{s_i} \\ \dot{\delta z}_{s_i} \end{pmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{500} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{1500} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \delta R_{cl} \\ \delta R_{trop} \\ \delta R_{ion} \\ \delta R_{selk} \\ \delta x_{s_i} \\ \delta y_{s_i} \\ \delta z_{s_i} \end{pmatrix} + \begin{pmatrix} w_{cl} \\ w_{trop} \\ w_{ion} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.22)$$

with initial covariances given by

$$\mathbf{P}_{GPS}(t_0) = \begin{bmatrix} 0.25 ft^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 ft^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.0 ft^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 25 ft^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25 ft^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 25 ft^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 25 ft^2 \end{bmatrix} \quad (3.23)$$

and means and noise strengths:

$$E \{ \mathbf{w}_{GPS}(t) \} = \mathbf{0} \quad (3.24)$$

$$E \{ \mathbf{w}_{GPS}(t) \mathbf{w}_{GPS}^T(t + \tau) \} = \begin{bmatrix} 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.004 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.004 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} ft^2/sec \cdot \delta(\tau) \quad (3.25)$$

3.3.3.2 GPS Measurement Model. The pseudorange measurements between the user and the SV's is shown in Figure 3.3 as R_{GPS} . This measurement is the sum of the true range, several error sources and a random noise:

$$R_{GPS} = R_t + \delta R_{cl} + \delta R_{trop} + \delta R_{ion} + \delta R_{sclk} + \delta R_{ucik} + v \quad (3.26)$$

where

R_{GPS}	=	GPS pseudorange measurement, from SV to user
R_t	=	true range, from SV to user
δR_{cl}	=	range error due to code loop error
δR_{trop}	=	range error due to tropospheric delay
δR_{ion}	=	range error due to ionospheric delay
δR_{sclk}	=	range error due to SV clock error
δR_{ucik}	=	range error due to User clock error
v	=	zero-mean white Gaussian measurement noise

Because R_t is not available to the filter, a substitution will be made to eliminate this term through the same techniques used to solve for the RRS measurement equation. First, the satellite position vector \mathbf{X}_s and the user position vector \mathbf{X}_u are defined as:

$$\mathbf{X}_u = \begin{bmatrix} x_u \\ y_u \\ z_u \end{bmatrix}^e, \quad \mathbf{X}_s = \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix}^e \quad (3.27)$$

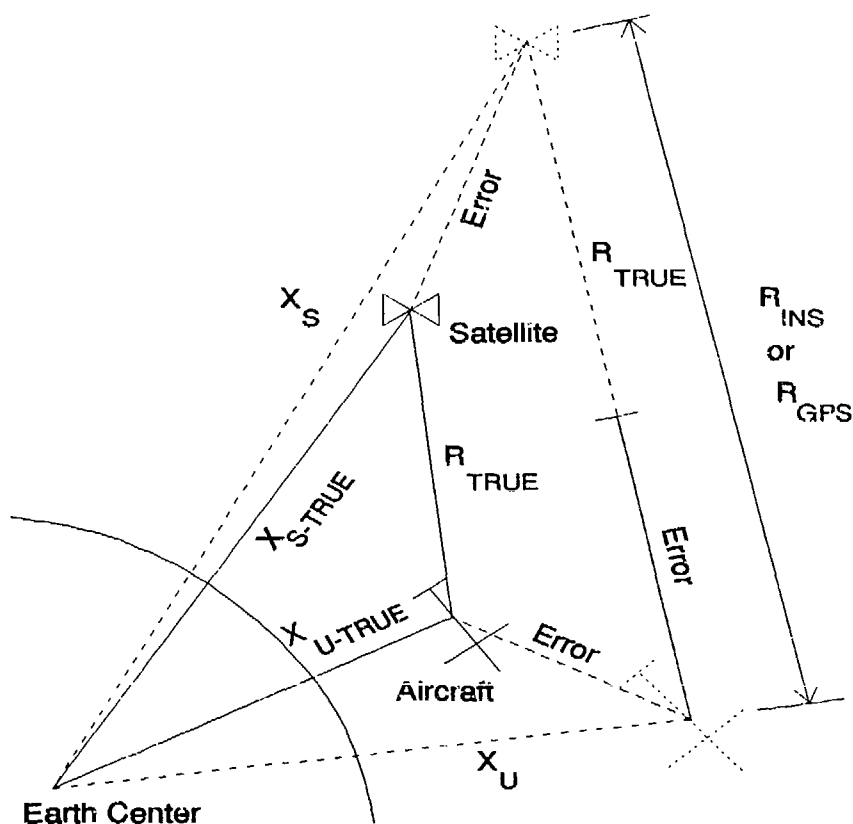


Figure 3.3. GPS Measurements

then the pseudorange from the user to the satellites is calculated by the INS as:

$$R_{INS} = \left| \mathbf{X}_U - \mathbf{X}_S \right| = \left| \begin{Bmatrix} x_U \\ y_U \\ z_U \end{Bmatrix}^e - \begin{Bmatrix} x_S \\ y \\ z_S \end{Bmatrix}^e \right| \quad (3.28)$$

An equivalent form for Equation (3.28) is:

$$R_{INS} = \sqrt{(x_U - x_S)^2 + (y_U - y_S)^2 + (z_U - z_S)^2} \quad (3.29)$$

Based on assumption 11 with perturbations representing the errors in \mathbf{X}_U and \mathbf{X}_S , Equation (3.29) can be approximated and written in terms of the true range and a truncated first-

order Taylor series:

$$\begin{aligned}
 R_{INS} = R_t &+ \left. \frac{\partial R_{INS}(\mathbf{X}_S, \mathbf{X}_U)}{\partial \mathbf{X}_S} \right|_{(\mathbf{X}_S, \mathbf{X}_U)_{nom}} \cdot \delta \mathbf{X}_S \\
 &+ \left. \frac{\partial R_{INS}(\mathbf{X}_S, \mathbf{X}_U)}{\partial \mathbf{X}_U} \right|_{(\mathbf{X}_S, \mathbf{X}_U)_{nom}} \cdot \delta \mathbf{X}_U
 \end{aligned} \quad (3.30)$$

The solution for R_{INS} is found by substituting Equation (3.29) into Equation (3.30) and evaluating the partial derivatives to get:

$$\begin{aligned}
 R_{INS} = R_t &- \left[\frac{x_s - x_u}{|R_{INS}|} \right] \cdot \delta x_u - \left[\frac{y_s - y_u}{|R_{INS}|} \right] \cdot \delta y_u - \left[\frac{z_s - z_u}{|R_{INS}|} \right] \cdot \delta z_u \\
 &+ \left[\frac{x_s - x_u}{|R_{INS}|} \right] \cdot \delta x_s + \left[\frac{y_s - y_u}{|R_{INS}|} \right] \cdot \delta y_s + \left[\frac{z_s - z_u}{|R_{INS}|} \right] \cdot \delta z_s
 \end{aligned} \quad (3.31)$$

Finally, the GPS pseudorange *difference* measurement is given as:

$$\begin{aligned}
 \delta z &= R_{INS} - R_{GPS} \\
 &= - \left[\frac{x_s - x_u}{|R_{INS}|} \right] \cdot \delta x_u - \left[\frac{y_s - y_u}{|R_{INS}|} \right] \cdot \delta y_u - \left[\frac{z_s - z_u}{|R_{INS}|} \right] \cdot \delta z_u \\
 &+ \left[\frac{x_s - x_u}{|R_{INS}|} \right] \cdot \delta x_s + \left[\frac{y_s - y_u}{|R_{INS}|} \right] \cdot \delta y_s + \left[\frac{z_s - z_u}{|R_{INS}|} \right] \cdot \delta z_s \\
 &- [1] \delta R_{cl} - [1] \delta R_{trop} - [1] \delta R_{ion} \\
 &- [1] \delta R_{Sclk} - [1] \delta R_{Uclk} - v
 \end{aligned} \quad (3.32)$$

3.3.3.3 The Reduced Two-State GPS Filter Model. Various research efforts have shown that two states provide a sufficient model for a GPS (10, 23). The primary argument is that the errors modeled by the other 28 states are small when compared to states one and two which are common to all SV's. By adding measurement noise (increasing \mathbf{R}) and retuning the filter, the overall performance of the NRS can be maintained with a significantly reduced order model. The final noise values are shown in Tables B.10 and B.11.

3.4 Failure Models

This section discusses the methods used to model failures in the MSOFE simulations and the corresponding models used by the FDI algorithm for detection.

3.4.1 Simulation Failure Models. The 10 different types of failures that were modeled are presented with actual values in Table 3.1. A short description of each failure is as follows:

1. **Jamming** – Jamming is modeled as a sudden increase or jump in the measurement noise associated with all four SV's. This failure is induced in all SV signals because the jamming is assumed to occur at the receiver, which will affect all four channels simultaneously. The jamming noise, RS , is added to the system model measurements only, with values shown in Table 3.1. These values represent various levels of jamming which result in lower carrier-to-noise ratios, C/N_0 , of the GPS signal and were calculated using:

$$\frac{RS}{\Delta^2} = \frac{K_1 B_n}{(C/N_0)} + \frac{K_2 B_i B_n}{(C/N_0)^2} \quad (3.33)$$

where

RS = jamming noise added to system measurement noise

Δ = code modulation chip width

C/N_0 = carrier-to-noise ratio

B_n = code tracking loop noise bandwidth

B_i = one-sided IF bandwidth

K_1 = code mechanization parameter constant

K_2 = code mechanization parameter constant

The theory supporting Equation (3.33) can be found in an article by Martin (12).

Table 3.1. Failure Types and Models

Run	Goal	Fail Type	Method	Time Frame	Comments
0	Baseline	No Failure	N/A	N/A	
1	Heavy Jamming	Jump in Measurement Noise	Increase R from 2 to 4000	2000-4000 sec	$C/N_0 = 15$ dB-Hz
2	Medium Jamming	Jump in Measurement Noise	Increase R from 2 to 500	2000-4000 sec	$C/N_0 = 20$ dB-Hz
3	Light Jamming	Jump in Measurement Noise	Increase R from 2 to 30	2000-4000 sec	$C/N_0 = 25$ dB-Hz
4	Spoofing	Bias on ONE Measurement	Add bias=7000 to SV1	2000-4000 sec	Causes 1 mi position error
5	Spoofing	Bias on ONE Measurement	Add bias=700 to SV1	2000-4000 sec	Causes 500 ft position error
6	Spoofing	Bias on ONE Measurement	Add bias=700 to SV1	4000-6000 sec	Causes 500 ft position error
7	Spoofing	Bias on ALL Measurements	Add bias=700 to SV1-SV4	2000-4000 sec	Causes small position error
8	Spoofing	Ramp on ONE Measurement	Add ramp=2T to SV1	1500-6000 sec	Causes 1 1/2 miles position error
9	Spoofing	Ramp on ONE Measurement	Add ramp=1T to SV1	2000-3000 sec	Causes 700 ft position error
10	GPS Fail	Loss of ONE Measurement	SV1=0 over time frame	2000-2200 sec	Causes complete disruption

2. Spoofing

- (a) Model A – A bias is added to the measurements associated with one or all four SV's designated SV1–SV4. This bias will be added suddenly in time and the values chosen represent a variety of spoofers. An intelligent spoofer will have an accurate estimation of the measurements being received by the aircraft from the satellites and can *discretely* add an undetectable bias to these measurements. A less sophisticated spoofer would have to use a larger bias to ensure effectiveness in corrupting the measurements while running the risk of being detected. A simple method was used to determine the amount of bias added to the pseudorange and is illustrated through an example. Assume the spoofer wanted to pull the aircraft approximately one mile off course in terms of latitude and longitude. The net *distance error* would be the magnitude of the change in the filter computed latitude and longitude error with and without the spoofer:

$$\text{Distance error} = \sqrt{(\delta\theta_x - \delta\theta_{x:\text{spoofer}})^2 + (\delta\theta_y - \delta\theta_{y:\text{spoofer}})^2} \quad (3.34)$$

where

$$\begin{aligned} \delta\theta_{x:\text{spoofer}} &= \text{latitude error with spoofing} \\ \delta\theta_{y:\text{spoofer}} &= \text{longitude error with spoofing} \end{aligned}$$

The altitude channel is not used to compute the distance error because altitude information is readily available from other instruments and the goal is to draw the aircraft off target. Figure C.3 shows the *increase* in the filter-computed errors in all three channels and the resulting distance error of approximately one mile over the time frame of the failure for a bias of 7000 ft on SV1. Only one of the simulation runs will induce the bias in more than one SV signal. It is considered unrealistic that a spoofer would be able to identify all four SV's selected by the GPS receiver consistently because the SV's are chosen based on their geographic relation to the receiver. The spoofer would not only be more complex, but would require four receiver/transmitters to accomplish this task.

Finally, simultaneously adding a bias to all four SV measurements would appear as an increase in the user clock bias error and would be quickly compensated by the filter. In contrast, a bias added to only one of the four GPS channels should cause errors in the navigation solution and degrade filter performance throughout the time frame of the failure. A single run will be performed to verify these predictions and show the distinction of these failure types.

(b) Model B – An even more subtle failure would be a ramped value added to the SV measurements. The ramp rates are somewhat arbitrary but with the same basic idea of slowly drawing the aircraft off course. Slopes of one ft/sec and two ft/sec are simulated with net position errors of 700 ft after 1000 sec and 1 1/2 miles after 4500 sec, respectively. Simulations were not performed with a ramp value added to all the measurements as explained above.

3. GPS failure – Represented as a loss of one or all SV measurements and modeled by setting the measurement $z=0$. The time frame illustrates that the SV will be lost for a period of time but may be reacquired. Simulations with all four SV signals removed will not be done, as failing one channel has the same effect of completely disrupting the filter.

3.4.2 FDI Failure Models. The multiple GLR algorithm allows for various hypotheses based on the types of failures being detected. A different failure model can be formed for each matching filter to optimize detection and isolation of failures through observation of the MLE's and GLR's associated with each matching filter. In simpler terms, the matching filter with the largest GLR is deemed the most correct in finding the failure. Given a group of different failures, a matching filter could be generated based on each failure type or a less complex FDI scheme would be to model only the most likely failures. By selectively choosing a bank of matching filters to cover most of the failures, the computational burden on the computer running the FDI algorithm can be reduced from one running an excessive number of filters in parallel. This concept is used in conjunction with a simple chi-square test to form a two-level FDI scheme. The first level consists of both the chi-square test and five matching filters to provide initial detection of a failure.

The second level is used to isolate the failure based on the five GLR outputs alone. This scheme benefits from the simplicity and reliability of the chi-square test and the flexibility of the multiple GLR tests.

The failure quantities in Equation (2.30) are specifically defined to represent the matching filter models. The function $n(t_i, \theta)$ is equal to one for all t_i , which implies that the failure occurs at the beginning of the window (see assumption 12). The failure vector $\mathbf{d}(t_i)$ is defined in general as

$$\mathbf{d}(t_i) = \begin{bmatrix} SV1 \\ SV2 \\ SV3 \\ SV4 \\ TR1 \\ TR2 \\ TR3 \\ TR4 \\ TR5 \\ TR6 \end{bmatrix} \quad (3.35)$$

where SV1 - SV4 and TR1 - TR6 represent the satellite vehicles and transponders respectively and will have a value of zero or one for this simplified model. Note that the Doppler and altimeter measurements are not included in these models because their measurement residuals are virtually unaffected by the failures in question and they slow down the algorithm when included in these equations. The plots of the GLR test using all 14 measurements (Doppler and altimeter included) are not presented because they look essentially the same as the GLR test using only 10 measurements.

The first four matching filters assume a failure in only one of the four SV's and the last filter models a failure in all SV's simultaneously, resulting in failure vectors of the form:

$$\begin{array}{ccccc}
 \text{SV1} & \text{SV2} & \text{SV3} & \text{SV4} & \text{SV1-4} \\
 \\
 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} & , & \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} & , & \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} & , & \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} & \text{or} & \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
 \end{array} \quad (3.36)$$

These models should perform well on bias type failures and encounter some delay in detecting ramp type failures. This is expected since the matching filters are designed for biases and not ramps, but the ramp results will show how well the filters perform on other types of failures. Further studies could include more complex models specifically designed to detect ramp failures.

3.5 Summary

This chapter presented the details for both the navigation filter and failure models. The basis for the measurement models was discussed to help describe the intricacies of the NRS design. The state and dynamics model descriptions illustrate the high degree of nonlinearity and time-variance of the system. The reduced order filter models were presented, including consideration of the critical job of tuning the NRS filter. The methods used to induce the failures in the simulations were shown along with the models for the matching filters designed to detect and isolate these failures. Results and analysis of these simulations are presented in the next chapter.

IV. Results and Analysis

4.1 Assumption Verification

One of the primary concerns in this research has been to verify assumption 6 presented in Chapter I. The GLR algorithm required a discretized version of the state dynamics matrix \mathbf{F} which was assumed piecewise constant over the two-second sample period. All 43 time varying elements of \mathbf{F} were carefully scrutinized to ensure that they did not change significantly between samples. Figure C.2(a) shows the worst case element of \mathbf{F} in terms of time variance over the entire simulation. From this plot it is hard to see if the high dynamics are being preserved over the sample period. Figure C.2(b) provides a closer look at the changes occurring in this plot during the most dynamic time period. The time scale in Figure C.2(b) is reduced so that values at every two-second sample are clearly seen. From this plot it is obvious that the quickly changing nature of the \mathbf{F} matrix is preserved with a sample time of two seconds, so the discretized STM, Φ , based on assumption 6 will be valid.

4.2 NRS Filter Performance

A variety of simulations were run on MSOFE to determine the final tuning values chosen for the filter. Observation of state statistics and measurement residuals gave insights into adjustments of the tuning values necessary to provide good state estimation while enhancing FDI potential as discussed in Section 2.3.3. The state plots in Appendix D represent a well tuned filter with the primary concern being to reduce the mean error for each state. A quick method for evaluating how well the filter is tuned is to ensure that the mean \pm sigma for the state error is bounded by the filter-computed zero \pm sigma. The FDI potential is evaluated using the residual plots and the goal is to bound the residual \pm sigma by the filter-computed zero \pm sigma derived from the residual covariance. The baseline plots shown in Figures D.16-D.20 indicate that further reduction of the measurement noise covariance, \mathbf{R} , would improve the FDI performance based on $\mathbf{\Lambda} = \mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R}$. However, lower values of \mathbf{R} caused serious degradation of filter tuning for state estimation performance. It is also unrealistic for \mathbf{R} to have values below the actual measurement

device noise variance values. Therefore, a tradeoff was made resulting in final values for \mathbf{Q} and \mathbf{R} shown in Tables B.10 and B.11.

The 15-state filter performance is compared against the results of the thesis by Negast (23) and the operational specifications listed in Section 1.6.4. All results are based on 10-run Monte Carlo simulations with ensemble averaging performed over the 10 runs. Negast conducted studies on a variety of reduced order filters with some of his results shown in Table 4.1. The values presented are the temporal averages of the ensemble averages of true filter estimation errors (1σ) for the position, velocity, and attitude errors over the two-hour flight profile. Both the 97- and 69-state filters were evaluated against a 128-state truth model defined by Negast. The 15-state filter used in this thesis was evaluated against the 97-state truth model defined earlier. Although this disparity could slightly skew the results, a comparison of the filter performance is still worthwhile in demonstrating the ability of the 15-state filter to provide a good navigation solution.

Table 4.1. Temporal Averages of True Filter Errors (1σ)

Filter	Latitude (ft)	Longitude (ft)	Altitude (ft)	East Vel (fps)	North Vel (fps)	Up Vel (fps)	East Tilt (arcs)	North Tilt (arcs)	Azimuth (arcs)
Desired	13.00	13.00	40.00	0.100	0.100	0.400			
97-NRS	1.35	2.71	5.28	0.014	0.010	0.045	1.07	1.29	9.74
69-NRS	3.28	4.21	9.04	0.033	0.026	0.070	1.37	2.11	18.42
15-NRS	4.00	7.22	8.63	0.032	0.029	0.070	3.13	4.49	17.57

All the operational specifications were easily met and exceeded as shown in Table 4.1. It is clear from the results that the overall performance of the 15-state NRS filter is degraded in comparison to the higher order filters. Some of the errors are actually better for the 15-state filter than for the 69-state filter as a result of either the difference in truth models or good filter tuning. Investigation into the individual state plots shown in Appendix D reveals other problems encountered with the 15-state filter. Large variations in the true system behavior such as high dynamic maneuvers will test the robustness of the filter. With a reduced order filter like the 15-state NRS, a single sharp turn or altitude change is easily handled, but several of these maneuvers caused the filter estimation error

to increase significantly. The flight profile shown in Figure C.1 indicates that problems may occur between 4800 and 6600 seconds. Analysis of the filter performance verified that two or three high-g turns in rapid succession provided justification for increasing the tuning values of certain states in order to enhance tracking. Adaptive tuning techniques were employed on states one, two, 12 and 14 with upper and lower process noise values shown in Table B.10. The upper noise values would be employed following a series of quick turns and the lower noise values would be used again a few hundred seconds after the last major turn. A study would have to be done with a variety of flight profiles to determine the optimal conditions for selecting the upper or lower tuning values. The key issue is that adaptive tuning is required and was implemented for this simplified filter model as shown in Figures D.1, D.2, D.12 and D.14. Statistics for the other 11 states are plotted in Appendix D and indicate that all the states were estimable at various degrees of tuning. State 3 represents the azimuth error and, as anticipated, was difficult to estimate in this wander-azimuth system. Also, its magnitude is very small resulting in little or no impact on the NRS filter performance so fine tuning was not attempted.

Another goal was to verify the validity of the two-state RRS filter model. Although temporal averages were not presented by Negast for his 26-state RRS filter, visual examination of his plots reveals RRS range bias errors of 0.7 to 1.0 ft in comparison to 6.27 ft of error with the two-state model. This seemingly large increase had a small effect on overall filter performance and the two-state model is deemed to be sufficient for most applications. The major impact is the reduction in computational loading realized by removing 24 states from the filter.

4.3 FDI Performance

Both the GLR and chi-square FDI algorithms are discussed with run numbers correlating to Table 3.1. Specific concern is given for missed and false alarms and methods for enhancing failure detection are discussed. Three types of plots are discussed to illustrate the performance of the FDI algorithms. First, likelihood function plots show the actual results of the GLR algorithm. Second, a *fail flag* routine was written to smooth the GLR plots based on a detection threshold and a parameter called *number_low*. This routine

assumes there is a failure (*fail flag*=one) until the GLR value falls below the threshold for a number of successive samples \geq *number_low*, resulting in *fail flag*=0. The consequence of a large *number_low* is that *fail flag*=one for several samples even after the failure has been removed. Chi-square plots are also generated for three different window sizes. Techniques employing two *fail flag* routines with a large and small *number_low* could help make quicker decisions in dropping the *fail flag*. Other decision logic might be used to overcome problems in detecting certain types of failures, but this thesis focused on a simple scheme using a single *fail flag* routine and a single chi-square test for decisions.

4.3.1 Jamming Detection. Three levels of jamming were induced as indicated by runs one, two and three in Table 3.1. The degradation in the filter performance is directly related to the amount of jamming noise induced. Selected state plots are shown in Appendix E to illustrate the effects of heavy jamming on the filter. The filter is able to reacquire a good estimation of the states quickly following removal of the jamming noise, demonstrating the ability of the filter to survive a hostile environment. The residual plots shown in Figures E.4 and E.5 make it clear that simple residual monitoring would be sufficient to detect the jamming when compared with the baseline residuals in Figures D.16 – D.17. Figure E.6 seems to indicate that the transponders are not significantly affected by the jamming, but even these minor variations from the baseline in Figure D.18 will be detectable by both the GLR and chi-square algorithms.

The results of the GLR test for heavy jamming are shown in Figures E.7 and E.8 and should be compared to the baseline plot in Figure D.21. The five plots in each figure represent the GLR's based on the five different matching filters discussed in Section 3.4.2. As expected, the matching filter assuming a failure in all four satellites provides the best detection. Figure E.8 shows that with a threshold=120 and a *number_low*=23, the *fail flag* remains up (equal to one) throughout the failure time and missed alarms can be avoided. The relatively large value for *number_low* is attributed to the continuous variations in the residuals resulting from the addition of random noise. These variations prevent the GLR algorithm from reaching a constant level above the noise floor and the large *number_low*

causes an excessive 46-second delay, which is excessive, in determining that the jamming is removed.

Comparison of Figures D.21 and E.9 show that the chi-square test was very successful in detecting the jamming. The best choice for the window size is 15 samples in order to reduce delays in detection and false alarms which are both apparent at the end of the failure time. A larger window size would only be necessary if the threshold had to be set so low (for good detection) that it caused false alarms due to the noise floor.

Similar results were obtained with the medium jamming level, as shown in Figure E.10. Both the GLR and χ algorithms are capable of detecting this level of jamming. Neither algorithm was successful at consistently detecting the light jamming, as shown in Figure E.11. The main reason the GLR test had difficulties detecting jamming failures is that the matching filters are designed to isolate a bias failure and the jamming noise is quite different from a bias. In contrast, the chi-square test is not dependent on a failure model and is better suited to detect the variations in the residuals caused by jamming.

4.3.2 Bias Failures. Runs four – six induced bias failures on SV1 only. Figures F.1 and F.2 show the degradation in the filter throughout the time frame of the failure resulting from a bias of 7000 ft being added to the pseudorange of SV1. The GPS residuals shown in Figure F.3 clearly indicates that a failure begins at 2000 sec and falls off at 4000 sec. Although somewhat masked by the scale of the plot, Figure F.3(b) shows that the residual values are also large throughout the entire failure time. This is easier to see on the transponder residuals in Figure F.4. An interesting observation is the significant variation in the transponder residuals even though no failure was induced on their measurements. Similarly, all the SV residuals are affected as a result of a failure on SV1 only. This apparent *coupling* of the measurements is not due to correlation in the real world, but rather a result caused by the design of the NRS filter. The measurement models are functions of the error states within the filter and with large variations in these estimated states, all the measurement residuals are affected. This coupling of the measurements will have a negative effect on failure isolation. It is also important to recognize the large spike occurring at approximately 2300 sec, making the presence of the failure seem obvious.

This was determined to result from a sharp turn by the aircraft which dithered the system, indicating that FDI decisions should coincide in time with dithering. The significance of this dither signal will be discussed later.

A typical GLR plot is shown in Figure F.5. The lower plot has been capped off at 200 to allow viewing of the data close to the noise floor. This noise floor is the result of initial transients, variations in random noise and changes in the aircraft dynamics. A threshold could be set for quick detection and to prevent any false alarms (recognized as the GLR value crossing the threshold when no failure is present), but a missed alarm would result. The approach would be to run the *fail flag* routine with *number_low*=two, resulting in a two-second delay in realizing that the failure is gone, no missed alarms and a detection time of two seconds. This concept is applied repeatedly throughout the rest of the analysis.

The GLR values for all five matching filters are shown in Figure F.6. Analysis of this data generates *fail flag* results shown in Figure F.7. It is reassuring that the best results are obtained from the matching filter SV1 which assumes a bias failure in SV1. There is concern for the false alarms in SV2 and SV4. One solution would be to raise the threshold above the noise floor of both matching filters. Although no increase in detection delay will occur for this failure, later simulations with ramp failures will suffer from raising the threshold too high and causing delays in detection. A second more desirable method would be to rely on the chi-square test for detection. Figure F.9 shows favorable chi-square results in terms of clean and quick detection. A window size of 15 samples will give fast detection (two seconds), no false alarms and less than 20 sec of delay in dropping the failure condition.

The next problem is to isolate the failure. The chi-square test provides no indication of which sensor has failed, but the GLR results are much more useful. The GLR for SV1 has a very large value at 2300 sec, which matches the time of the dither discussed earlier, and this large spike distinguishes it from the other matching filters (see Figure F.6). A delay of almost 300 sec would exist, which is unacceptable, but clear identification of the failed sensor is realized based on a second threshold designed to isolate failures. Further discussion of this idea will be presented in Chapter V.

The follow-on goal of generating corrective feedback to the Kalman filter was not pursued extensively, but a quick look at the MLE of ν for this run provided some interesting results. Figure F.8 shows two plots of $\hat{\nu}$ with different vertical scales. The actual failure size was 7000 ft, and close inspection of these plots reveals that $\hat{\nu}$ could provide a crude estimate of the size of the failure during portions of the failure time. The same problem of poor performance prior to the dither signal is encountered here, along with disturbances in $\hat{\nu}$ after 3800 sec. The aircraft dynamics (an altitude change at 3800 sec) are speculated to have caused this disturbance. Continued research into using MLE's for corrective feedback is encouraged but these values must be checked carefully to ensure they do not adversely affect the filter performance by providing erroneous feedback.

Similar results were obtained for run five in which the bias was reduced to 700 ft. A single GLR plot based on SV1 is shown in Figure F.10 along with its corresponding *fail flag* result. A problem was encountered with missed alarms prior to the dither signal. Although variations in the threshold value and *number_low* could overcome this problem, these variations would inhibit detection of other types of failures discussed previously. The same solution to use the chi-square test for detection was available and it was determined that a threshold=50 would be effective for this failure (see Figure F.11). Other threshold values would also be acceptable for this failure, but using a threshold=50 coincided with other results. Final implementation studies could be accomplished to determine the best threshold value for all types of failures being considered.

In order to verify the influence of dither signals on the FDI algorithms, run six was conducted with the failure induced during the highly dynamic time frame from 4000 - 6000 sec. Large variations in the GLR and χ data correlated to quick turns in the flight profile, as anticipated. The GLR plot and *fail flag* based on SV1 are shown in Figures F.12 and F.13 with unfavorable results occurring with *number_low*=two. In order to prevent missed alarms, a *number_low*=45 was required. Again, a delay occurred in dropping the failure condition, but the chi-square test provided the necessary alarm with the same threshold=50 used earlier. The same technique of detection through the chi-square test and isolation via the GLR test would work for this failure.

The last bias type failure, run eight, assumed that all the SV's would be spoofed simultaneously. The state plots in Figures F.15 - F.17 show the robustness of the filter to adapt to this failure. Fewer than ten updates at a two-second sample rate were required for the filter to regain accurate tracking of all states. This coincides with the speculation in Section 3.4.1 that the filter would attribute this failure to an increase in the user clock bias. Figure F.18 shows that the failure is quickly detected but the adaptive nature of the EKF prevents a sustained recognition of the failure. Further simulations were conducted with various biases and values as low as 50 ft were easily detected by the GLR algorithm. This failure was not pursued further, for reasons explained in Section 3.4.1.

4.3.3 Ramp Failures. Runs eight and nine refer to ramp type failures of slopes two and one ft/sec, respectively. The state plots in Figures G.1 and G.2 show the gradually increasing degradation in the filter performance due to this type of spoofing. The residuals in Figures G.3 and G.4 indicate that failure detection should be possible, and the spikes in Figure G.3 highlight the effects of dithering caused by high dynamics.

The GLR algorithm was somewhat successful in detecting the larger ramp failure= $2T$ with some delay. Figure G.5 illustrates that the all five matching filters will cause missed and false alarms. A closer look at these results is shown in Figure G.6 in which the vertical scale has been reduced. From this plot the matching filters do not appear to provide quick or reliable detection. The results of the *fail flag* routine are shown in Figure G.7 with the threshold= 120 as discussed on page 4-4 and an increase in *number_low*= 10 . This increase in *number_low* did not adversely effect previous results on other types of failures other than the expected delay in dropping the *fail flag*. The two major problems with these results are the 356-sec detection delay seen in even the best matching filter, SV1, and the multiple missed and false alarms seen in the other matching filters. A positive result was the large spike seen at the time of the dither (2300 sec) that would provide isolation of the failure on SV1.

Again, the chi-square test provided better results for detection with the ramp= $2T$. Figure G.8 shows reliable detection, based on a threshold= 50 , beginning 250 sec after the failure which occurred at 1500 sec. The 250-sec delay seems excessive, but the failure

is subtle and the aircraft is spoofed approximately 400 ft at the time of detection. The chi-square test also dropped the failure more quickly than the GLR test and did not suffer from the high dynamics between 5000 and 6000 sec.

Results with the ramp=1T are shown in Figures G.9 - G.11. The GLR test was more reliable for detection, as shown in Figure G.10, when compared to the ramp=2T results, but this should be attributed to the lack of dynamics during the failure time frame. The chi-square test was not as successful with this more subtle failure and suffered the same delay in detecting the failure as did the GLR. The dither signal is again the key to detection, and isolation was also possible based on the matching filter results. For this particular simulation with the dither signal occurring 300 sec after the failure, the aircraft would only be spoofed approximately 150 ft off course prior to detection. Although these errors due to detection delay might be acceptable for most missions, it points out the need to conduct a study on optimal input signals for the purpose of enhancing failure detection and isolation. Mehra discusses a variety of techniques for determining these optimal inputs including observation of the systems eigenvalues (18, 19). Some quick and simple methods for system identification are presented by Zarrop and Goodwin based on minimizing a scalar function of the information matrix (32). Consideration should also be given to developing failure models for the matching filters designed to detect ramp failures with the expectation that detection time will decrease.

4.3.4 Failed GPS. A complete loss of one pseudorange entering the NRS filter was induced, and the degraded performance of the filter is shown in Figure H.1. The filter loses track until 3800 sec after the failure is removed, revealing a major liability in the design of the NRS filter. Figures H.2 and H.3 show the residuals as excessive and divergent, indicating that the filter is unable to provide a good navigation solution in the absence of even one pseudorange (PR). One solution to this problem is, after detecting the loss of the PR causing the filter divergence, move the filter back in time by recalculating the Kalman filter equations without the PR measurement input. This would prevent the other measurements (transponders, baro-altimeter, and Doppler) from being affected by the coupling discussed earlier, and the filter would remain stable. Once the PR input was

reacquired, this measurement would be processed by the Kalman filter as before. This idea was not pursued due to time constraints but is mentioned in the recommendations for future work. The FDI results are skewed by the filter divergence and the GLR algorithm goes unstable. Detection of this failure is not a major concern since standard GPS receiver design would recognize the problem and compensate if possible. Finally, this type of failure would not be a good tactic for a spoofer, as discussed in Section 3.4.1, but it is comforting to know that this failure is distinguishable from typical spoofer-induced failures.

4.3.5 Numerical Precision and Modelling. Considerable attention was given to the numerical precision of the computers running the simulations and the FDI algorithms. Investigation into the matrices passed from MSOFÉ to Matrix_x revealed a potential problem. The two rows of the measurement matrix, \mathbf{H} , corresponding to states one and two were 10^8 larger than many of the other rows, but both computing systems were using double precision in all their calculations, so there might not be a problem. The positive performance of both the Kalman filter within MSOFÉ and the GLR algorithm in Matrix_x indicated that everything was fine, but it is possible that their performance could be improved by performing a similarity transformation on the filter model to scale the states more evenly. Details for this type of transformation are shown by Maybeck (13:p. 28). Time limitations prevented this work from being completed.

Another factor affecting the performance of the GLR test is mismodelling even though the NRS filter was shown to exceed specifications in providing a navigation solution. The dependence of the GLR algorithm on the system dynamics makes it susceptible to inaccuracies in modelling, especially when the truth model has 97 states versus the filter model of only 15 states. A quick study was performed in which both the truth and filter models had only 15 states to ensure that modelling was not a major factor in the GLR performance. No noticeable improvement resulted from this study. Detection delays on subtle failures were not significantly reduced and failure isolation was still not possible prior to the dither signals. These findings indicated that higher order models for the filter would not improve FDI performance sufficiently to justify the increase in computational loading required for a larger filter.

4.4 *Summary*

This chapter focussed on the results obtained in the research. A brief discussion of the piecewise-constant nature of the dynamics matrix and concerns about numerical precision and modelling indicated that these assumptions were valid. Results on filter performance were very good and a comparison to past research showed that the reduced order filter model worked well. Analysis of the FDI performance revealed some positive results but techniques for enhancing this performance would be necessary prior to final implementation on actual aircraft. Chapter V offers some suggestions to accomplish this goal and provides some ideas for future research.

V. Conclusions and Recommendations

This chapter provides a brief summary of the results, including a possible method of implementation for the FDI system. Recommendations for future research are also presented.

5.1 NRS Filter Performance

The 15-state NRS filter exceeded all operational specifications in its ability to provide an accurate navigation solution. Although the filter's performance was degraded as compared to higher order models used in prior research, the state estimates were reliable and accurate. Increases in tracking errors were experienced following a series of harsh maneuvers by the aircraft, prompting the use of adaptive tuning techniques on states one, two, 12 and 14. A simple technique required these states to have lower and upper process noise values for adaptation.

The reduced order RRS filter model provided adequate estimation for the transponders while significantly decreasing the number of states in the filter from 26 to two. The importance of this simplification is the reduction in the load on the computer tasked to run the Kalman filter algorithm. This accomplishment is magnified by the fact that many of the small aircraft on-board computers in the Air Force inventory lack the capability to run a Kalman filter with over 60 states and a two-second update rate.

5.2 FDI Performance

The discretization process used on the state transition matrix assumed time invariance over the two-second sample period and was sufficient to generate accurate inputs for the GLR algorithm. This resulted from the combination of the two-second sampling period and the 8th order Padé approximation for the matrix exponential function. Longer sample periods or cruder approximation techniques might cause the discretization process to be degraded, resulting in improper modelling within the GLR algorithm.

A combination of the GLR and chi-square algorithms provided adequate detection and isolation for most of the failures considered. Heavy and medium jamming levels were

reliably detected but light jamming was not noticable above the noise floor. Although jamming detection using the FDI algorithms was desired, many other forms of detection are available using current electronic warfare techniques. Even more important is the distinction between the outputs of the GLR matching filters when compared to each other and also when compared to other types of failures such as spoofing. It is necessary for the FDI system to recognize the type of failure prior to generating any corrective feedback into the NRS filter and to alarm the aircraft pilot properly of the hostile environment.

The FDI algorithms performed well for both types of spoofing. Thresholds were found empirically with substantial tradeoffs required to prevent false and missed alarms. All bias values were easily detected and isolated using a combination of the chi-square test for detection and the GLR test for isolation. The chi-square test minimized detection delay to two sec, eliminated false alarms and minimized the delay in dropping the failure condition to 20 sec. If the FDI algorithms were attempting to detect failures in a flight control system rather than a navigation system, then these delays would be unacceptable, but for the scenario described in Chapter I, these delays should not cause major problems in completing mission objectives. Ramp failures presented additional problems for the chi-square test. The gradual changes in the residuals caused delays in detection with larger ramp values having delays of about four minutes and small ramp values completely avoiding detection prior to dithering the filter. These results would not be acceptable for final implementation.

Two important aspects must be addressed for implementation of a reliable FDI system. First, a two-level scheme should be used in which the chi-square test is utilized for detection and the GLR test provides isolation. It should be realized that both algorithms use the inverse of the residual covariance, $\Lambda(t_i)$, in their calculations, so the chi-square test does not generate much computational overhead assuming the GLR test is already being conducted. The GLR algorithm does require a significant number of computations and may interfere with chi-square calculations done by an on-board computer, but the two-level design allows for priority to be given to the chi-square test for detection while the GLR test can run on a lower priority for isolation. A study on this idea could provide

optimal performance of the overall FDI scheme. Neither algorithm could independently outperform this two-level scheme in terms of detection and isolation.

The second major concern for implementation is whether purposely dithering the system could enhance FDI performance. Results clearly indicate that harsh aircraft maneuvers enhanced isolation for the GLR test on all failures and detection for the chi-square test on subtle or small failures. Maybeck discusses the use of probing with the intention that it "*purposely* excite certain modes of the system in order to *aid* the identification of uncertain parameters" (15:p. 229). One tactic discussed is the use of "S-turns" performed by the pilot and a simple variation in the flight profile would reveal the usefulness of this idea. Stratton, Menke and Hanlon looked into continuous and periodic dithering signals induced through control inputs (6, 20, 28). This would change Equation (2.1) to

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), t, \mathbf{u}(t)] + \mathbf{G}(t)\mathbf{w}(t) \quad (5.1)$$

where $\mathbf{u}(t)$ = control input. A study could be conducted on optimal control inputs for this navigation system to determine the best choices for this method. Preliminary indications are that rather large dither signals are required for this system so pilot-induced dithers would probably work better than automatically induced *subliminal* probes or automatically induced *nonsubliminal* probes that would bounce around the aircraft and its crew.

Comparison of a 15-state filter model against a 15-state truth model showed that higher order filter models would not significantly enhance FDI performance. Numerical precision was not considered to cause a major problem in the performance of the Kalman filter or the FDI algorithm, although re-scaling the states should still be considered.

5.2.1 Corrective Feedback. Although time constraints prevented research into feedback techniques designed to correct for failures, some observations can be made. As mentioned earlier, adaptive tuning techniques in which \mathbf{Q} was increased were employed to ensure adequate filter performance under normal conditions. This concept could be extended to enhance filter performance once a failure was detected. The FDI system is now confronted with a third task beyond that of detection and isolation. Assuming a bias failure has been induced, the FDI system must estimate the size of the bias in order to determine

the amount of adaptive feedback (represented by changes in \mathbf{Q}) needed to compensate for this failure. If the estimate is too large, the filter may become too conservatively tuned and performance will degrade. If the estimate is too small, insufficient tuning may occur and the filter will not be able to track the states closely. Additional problems may result from the filter becoming accustomed to the errors induced by the failure. If the filter adapts to the bias, it may become less aware of minor changes in the bias and might lose the ability to determine when the failure has been removed. Since the pilot would be alarmed of the initial failure, operator-induced dither signals could be used to help improve future detection and the overall filter performance would be improved.

Another method of compensation is to estimate the size of the failure and remove it from the incoming measurement. Assume the failure appears as a bias, b , on the measurement equation taken from Equation (2.11).

$$\delta \mathbf{z}_n(t_i) = \mathbf{H}[t_i; \mathbf{x}_n(t_i)] \delta \mathbf{x}_n(t) + \mathbf{v}(t_i) + b \quad (5.2)$$

If the GLR algorithm were able to find $\hat{\nu}$ accurately which is the MLE of the size of the failure, then simply subtracting $\hat{\nu}$ from Equation (5.2) would eliminate the failure in the system and the output of the Kalman filter would not be degraded. Other algorithms such as MMAE could also use this idea.

5.3 Recommendations

A brief list of recommendations for future research is presented, with many of the details concerning these items presented earlier.

- Finalize the similarity transform to rescale the states and compare results to those achieved without rescaling for both the Kalman filter and FDI algorithms.
- Verify dominance of the diagonal terms in the residual covariance matrix, $\Lambda(t_i)$, over the non-diagonal terms of this matrix. See assumption 7.

- Perform a study on optimal inputs intended to dither the system. Improvements should be noted in the NRS filter performance, but the main goal would be to decrease delays in detection and isolation of failures.
- Pursue techniques for feeding back corrective signals such as adaptive tuning or measurement correction discussed previously. Careful investigation of errors in the state estimates should indicate the success of these methods.
- Consider changing the failure models for the matching filters to look for ramp failures rather than biases. A scheme with 10 matching filters (five for biases AND five for ramps) should be compared against those with only five matching filters (five for biases OR five for ramps).
- Look into theoretical techniques for establishing thresholds rather than finding them empirically.
- Change the GLR algorithm listed in Appendix I to allow for various failure times. This implies that θ could take on any value within the window, resulting in an MLE of θ and a GLR as a function of t_i and $\hat{\theta}$. This GLR should better identify the time of the failure and may outperform the simplified GLR algorithm. Consideration would have to be given to increased detection time caused by the increased computations necessary to implement this new GLR test.
- Look into MMAE techniques as a replacement for the FDI scheme. MMAE could be used in conjunction with many of the recommendations listed above and the chi square test should always be considered as an additional source of information. The most significant task associated with this idea is developing the software capable of performing the simulations. MSOFE is not currently structured to handle the bank of Kalman filters associated with an MMAE scheme. The alternative would be to adopt another software package designed to run multiple Kalman filters. The major problem would be to convert the models and particularly the overhead associated with generating the measurements for the GPS and RRS portions of the NRS system. In either case, a head-to-head comparison of the results achieved by the MMAE method

and those achieved in this thesis should provide insights into the relative strengths and weaknesses of the GLR and MMAE approaches.

Appendix A. *Error Model State Definitions*

Tabular listings of the truth and filter models are presented. Tables A.1 and A.2 show the 41-state INS truth model with the LN-93 state numbers given for reference to the Litton technical report on the INS (9). Tables A.3 and A.4 list the RRS and GPS states respectively and Table A.5 lists only the states used in the NRS filter model.

Table A.1. 41-State INS System Model: First 20 States

State Number	State Symbol	Definition	LN-93 State
1	$\delta\theta_x$	X-component of vector angle from true to computer frame	1
2	$\delta\theta_y$	Y-component of vector angle from true to computer frame	2
3	$\delta\theta_z$	Z-component of vector angle from true to computer frame	3
4	ϕ_x	X-component of vector angle from true to platform frame	4
5	ϕ_y	Y-component of vector angle from true to platform frame	5
6	ϕ_z	Z-component of vector angle from true to platform frame	6
7	δV_x	X-component of error in computed velocity	7
8	δV_y	Y-component of error in computed velocity	8
9	δV_z	Z-component of error in computed velocity	9
10	δh	Error in vehicle altitude above reference ellipsoid	10
11	δh_B	Total baro-altimeter correlated error	23
16	δh_L	Error in lagged inertial altitude	11
17	δS_3	Error in vertical channel aiding state	12
18	δS_4	Error in vertical channel aiding state	13
19	∇_{x_c}	X-component of accelerometer and velocity quantizer correlated noise	17
20	∇_{y_c}	Y-component of accelerometer and velocity quantizer correlated noise	18
21	∇_{z_c}	Z-component of accelerometer and velocity quantizer correlated noise	19
22	δg_x	X-component of gravity vector errors	20
23	δg_y	Y-component of gravity vector errors	21
24	δg_z	Z-component of gravity vector errors	22

Table A.2. 41-State INS System Model: Second 21 States

State Number	State Symbol	Definition	I.N-93 State
25	b_x	X-component of gyro drift rate repeatability	30
26	b_y	Y-component of gyro drift rate repeatability	31
27	b_z	Z-component of gyro drift rate repeatability	32
28	S_{gx}	X-component of gyro scale factor error	33
29	S_{gy}	Y-component of gyro scale factor error	34
30	S_{gz}	Z-component of gyro scale factor error	35
31	∇_{bx}	X-component of accelerometer bias repeatability	48
32	∇_{by}	Y-component of accelerometer bias repeatability	49
33	∇_{bz}	Z-component of accelerometer bias repeatability	50
34	S_{Ax}	X-component of accelerometer and velocity quantizer scale factor error	51
35	S_{Ay}	Y-component of accelerometer and velocity quantizer scale factor error	52
36	S_{Az}	Z-component of accelerometer and velocity quantizer scale factor error	53
37	S_{QA_x}	X-component of accelerometer and velocity quantizer scale factor asymmetry	54
38	S_{QA_y}	Y-component of accelerometer and velocity quantizer scale factor asymmetry	55
39	S_{QA_z}	Z-component of accelerometer and velocity quantizer scale factor asymmetry	56
40	μ_1	X accelerometer misalignment about Z-axis	66
41	μ_2	Y accelerometer misalignment about Z-axis	67
42	μ_3	Z accelerometer misalignment about Y-axis	68
43	σ_1	X-accelerometer misalignment about Y-axis	69
44	σ_2	Y-accelerometer misalignment about X-axis	69
45	σ_3	Z-accelerometer misalignment about X-axis	69

Table A.3. 26-State RRS System Model

State Number	State Symbol	Definition
12	δR_b	Range error due to equipment bias
13	δv_b	Velocity error due to equipment bias
46	δP_{T1_x}	Transponder 1 x-component of position error
47	δP_{T1_y}	Transponder 1 y-component of position error
48	δP_{T1_z}	Transponder 1 z-component of position error
49	δR_{T1_a}	Transponder 1 range error due to atm propagation
50	δP_{T2_x}	Transponder 2 x-component of position error
51	δP_{T2_y}	Transponder 2 y-component of position error
52	δP_{T2_z}	Transponder 2 z-component of position error
53	δR_{T2_a}	Transponder 2 range error due to atm propagation
54	δP_{T3_x}	Transponder 3 x-component of position error
55	δP_{T3_y}	Transponder 3 y-component of position error
56	δP_{T3_z}	Transponder 3 z-component of position error
57	δR_{T3_a}	Transponder 3 range error due to atm propagation
58	δP_{T4_x}	Transponder 4 x-component of position error
59	δP_{T4_y}	Transponder 4 y-component of position error
60	δP_{T4_z}	Transponder 4 z-component of position error
61	δR_{T4_a}	Transponder 4 range error due to atm propagation
62	δP_{T5_x}	Transponder 5 x-component of position error
63	δP_{T5_y}	Transponder 5 y-component of position error
64	δP_{T5_z}	Transponder 5 z-component of position error
65	δR_{T5_a}	Transponder 5 range error due to atm propagation
66	δP_{T6_x}	Transponder 6 x-component of position error
67	δP_{T6_y}	Transponder 6 y-component of position error
68	δP_{T6_z}	Transponder 6 z-component of position error
69	δR_{T6_a}	Transponder 6 range error due to atm propagation

Table A.4. 30-State GPS System Model

State Number	State Symbol	Definition
14	δR_{clk_u}	User clock bias
15	δD_{clk_u}	User clock drift
70	δR_{loop_1}	SV 1 code loop error
71	δR_{trop_1}	SV 1 tropospheric error
72	δR_{ion_1}	SV 1 ionospheric error
73	$\delta R_{clk_{sv_1}}$	SV 1 clock error
74	δx_{sv_1}	SV 1 x-component of position error
75	δy_{sv_1}	SV 1 y-component of position error
76	δz_{sv_1}	SV 1 z-component of position error
77	δR_{loop_2}	SV 2 code loop error
78	δR_{trop_2}	SV 2 tropospheric error
79	δR_{ion_2}	SV 2 ionospheric error
80	$\delta R_{clk_{sv_2}}$	SV 2 clock error
81	δx_{sv_2}	SV 2 x-component of position error
82	δy_{sv_2}	SV 2 y-component of position error
83	δz_{sv_2}	SV 2 z-component of position error
84	δR_{loop_3}	SV 3 code loop error
85	δR_{trop_3}	SV 3 tropospheric error
86	δR_{ion_3}	SV 3 ionospheric error
87	$\delta R_{clk_{sv_3}}$	SV 3 clock error
88	δx_{sv_3}	SV 3 x-component of position error
89	δy_{sv_3}	SV 3 y-component of position error
90	δz_{sv_3}	SV 3 z-component of position error
91	δR_{loop_4}	SV 4 code loop error
92	δR_{trop_4}	SV 4 tropospheric error
93	δR_{ion_4}	SV 4 ionospheric error
94	$\delta R_{clk_{sv_4}}$	SV 4 clock error
95	δx_{sv_4}	SV 4 x-component of position error
96	δy_{sv_4}	SV 4 y-component of position error
97	δz_{sv_4}	SV 4 z-component of position error

Table A.5. 15-State Reduced-Order Filter Model

State Number	State Symbol	Definition
1	$\delta\theta_x$	X-component of vector angle from true to computer frame
2	$\delta\theta_y$	Y-component of vector angle from true to computer frame
3	$\delta\theta_z$	Z-component of vector angle from true to computer frame
4	ϕ_x	X-component of vector angle from true to platform frame
5	ϕ_y	Y-component of vector angle from true to platform frame
6	ϕ_z	Z-component of vector angle from true to platform frame
7	δV_x	X-component of error in computed velocity
8	δV_y	Y-component of error in computed velocity
9	δV_z	Z-component of error in computed velocity
10	δh	Error in vehicle altitude above reference ellipsoid
11	δh_B	Total baro-altimeter correlated error
12	δR_b	Range error due to equipment bias
13	δv_b	Velocity error due to equipment bias
14	δx_{clk_b}	User clock bias
15	$\delta x_{clk_{dr}}$	User clock drift

Appendix B. *Dynamics Matrices and Noise Values*

B.1 Definition of Dynamics Matrices

The LN-93 error-state dynamics matrix \mathbf{F} is defined in Chapter III as a combination of submatrices. The NON-ZERO elements of these submatrices are presented in the tables which follow. All the variables shown in the following tables are defined in the LN-93 technical report along with their units (9).

Table B.1. Elements of the Dynamics Submatrix F_{11}

Element	Term	Element	Term
(1,3)	$-\rho_y$	(1,8)	$-C_{RY}$
(2,3)	ρ_x	(2,7)	C_{RX}
(3,1)	ρ_y	(3,2)	$-\rho_x$
(4,2)	$-\Omega_z$	(4,3)	Ω_y
(4,5)	ω_{it_x}	(4,6)	$-\omega_{it_y}$
(4,8)	$-C_{RY}$	(5,1)	Ω_z
(5,3)	$-\Omega_x$	(5,4)	$-\omega_{it_x}$
(5,6)	ω_{it_x}	(5,7)	C_{RX}
(6,1)	$-\Omega_y$	(6,2)	Ω_x
(6,4)	ω_{it_y}	(6,5)	$-\omega_{it_x}$
(7,1)	$-2V_y\Omega_y - 2V_z\Omega_z$	(7,2)	$2V_y\Omega_x$
(7,3)	$2V_z\Omega_y$	(7,5)	$-A_z$
(7,6)	A_y	(7,7)	$-V_zC_{RX}$
(7,8)	$2\Omega_z$	(7,9)	$-\rho_y - 2\Omega_y$
(8,1)	$2V_x\Omega_y$	(8,2)	$-2V_x\Omega_x - 2V_z\Omega_z$
(8,3)	$2v_z\Omega_y$	(8,4)	A_z
(8,6)	$-A_x$	(8,7)	$-2\Omega_z$
(8,8)	$-V_zC_{RY}$	(8,9)	$\rho_x + 2\Omega_x$
(9,1)	$2V_x\Omega_z$	(9,2)	$2V_y\Omega_z$
(9,3)	$-2V_y\Omega_y - 2V_z\Omega_x$	(9,4)	$-A_y$
(9,5)	A_x	(9,7)	$\rho_y + 2\Omega_y + V_xC_{RX}$
(9,8)	$-\rho_x - 2\Omega_x + V_yC_{RY}$	(9,10)	$2g_o/a$
(9,16)	$-k_2$	(9,17)	-1
(9,18)	k_2	(10,9)	1
(10,16)	$-k_1$	(10,18)	$k_1 - 1$
(16,10)	1	(16,16)	-1
(17,16)	k_3	(17,18)	$-k_3$
(18,10)	k_4	(18,16)	$-k_4$
(18,18)	$k_4 - 1$		

Table B.2. Elements of the Dynamics Submatrix F_{12}

Element	Term	Element	Term	Element	Term
(7,19)	C_{11}	(7,20)	C_{12}	(7,21)	C_{13}
(7,22)	1	(8,19)	C_{21}	(8,20)	C_{22}
(8,21)	C_{23}	(8,23)	1	(9,19)	C_{31}
(9,20)	C_{32}	(9,21)	C_{33}	(9,24)	1
(9,11)	k_2	(10,11)	k_1	(17,11)	$-k_3$
(18,11)	$k_4/600$				

Table B.3. Elements of the Dynamics Submatrix F_{13}

Element	Term	Element	Term	Element	Term
(4,25)	C_{11}	(4,26)	C_{12}	(4,27)	C_{13}
(4,28)	$C_{11}\omega_{ib_x}$	(4,29)	$C_{12}\omega_{ib_y}$	(4,30)	$C_{13}\omega_{ib_z}$
(5,25)	C_{21}	(5,26)	C_{22}	(5,27)	C_{23}
(5,28)	$C_{21}\omega_{ib_x}$	(5,29)	$C_{22}\omega_{ib_y}$	(5,30)	$C_{23}\omega_{ib_z}$
(6,25)	C_{31}	(6,26)	C_{32}	(6,27)	C_{33}
(6,28)	$C_{31}\omega_{ib_x}$	(6,29)	$C_{32}\omega_{ib_y}$	(6,30)	$C_{33}\omega_{ib_z}$

Table B.4. Elements of the Dynamics Submatrix F_{14}

Element	Term	Element	Term	Element	Term
(7,31)	C_{11}	(7,32)	C_{12}	(7,33)	C_{13}
(7,34)	$C_{11}A_x^B$	(7,35)	$C_{12}A_y^B$	(7,36)	$C_{13}A_z^{B'}$
(7,37)	$C_{11} A_x^B $	(7,38)	$C_{12} A_y^B $	(7,39)	$C_{13} A_z^{B'} $
(7,40)	$C_{11}A_y^B$	(7,41)	$-C_{12}A_x^B$	(7,42)	$C_{13}A_y^B$
(7,43)	$C_{13}A_x^B$	(8,31)	C_{21}	(8,32)	C_{22}
(8,33)	C_{23}	(8,34)	$C_{21}A_x^B$	(8,35)	$C_{22}A_y^B$
(8,36)	$C_{23}A_z^{B'}$	(8,37)	$C_{21} A_x^B $	(8,38)	$C_{22} A_y^B $
(8,39)	$C_{23} A_z^{B'} $	(8,40)	$C_{21}A_y^B$	(8,41)	$-C_{22}A_x^B$
(8,42)	$C_{23}A_y^B$	(8,43)	$C_{23}A_x^B$	(9,31)	C_{31}
(9,32)	C_{32}	(9,33)	C_{33}	(9,34)	$C_{31}A_x^B$
(9,35)	$C_{32}A_y^B$	(9,36)	$C_{33}A_z^{B'}$	(9,37)	$C_{31} A_x^B $
(9,38)	$C_{32} A_y^B $	(9,39)	$C_{33} A_z^{B'} $	(9,40)	$C_{31}A_x^B$
(9,41)	$-C_{32}A_x^B$	(9,42)	$C_{33}A_y^B$	(9,43)	$C_{33}A_z^B$

Table B.5. Elements of the Dynamics Submatrix F_{22}

Element	Term	Element	Term	Element	Term
(19,19)	$-\beta_{\nabla_{x_c}}$	(20,20)	$-\beta_{\nabla_{y_c}}$	(21,21)	$-\beta_{\nabla_{z_c}}$
(22,22)	$-\beta_{\delta g_x}$	(23,23)	$-\beta_{\delta g_y}$	(24,24)	$-\beta_{\delta g_z}$
(11,11)	$-\beta_{\delta h_c}$				

The NON-ZERO elements of the dynamics matrix representing the the GPS and RRS are shown in the following table.

Table B.6. Elements of the Dynamics Matrix for GPS & RRS

(49,49)	$-1/300 \text{ ft}^2/\text{sec}$	(53,53)	$-1/300 \text{ ft}^2/\text{sec}$	(57,57)	$-1/300 \text{ ft}^2/\text{sec}$
(61,61)	$-1/300 \text{ ft}^2/\text{sec}$	(65,65)	$-1/300 \text{ ft}^2/\text{sec}$	(69,69)	$-1/300 \text{ ft}^2/\text{sec}$
(70,70)	$-1 \text{ ft}^2/\text{sec}$	(71,71)	$-1/500 \text{ ft}^2/\text{sec}$	(72,72)	$-1/1500 \text{ ft}^2/\text{sec}$
(77,77)	$-1 \text{ ft}^2/\text{sec}$	(78,78)	$-1/500 \text{ ft}^2/\text{sec}$	(79,79)	$-1/1500 \text{ ft}^2/\text{sec}$
(84,84)	$-1 \text{ ft}^2/\text{sec}$	(85,85)	$-1/500 \text{ ft}^2/\text{sec}$	(86,86)	$-1/1500 \text{ ft}^2/\text{sec}$
(91,91)	$-1 \text{ ft}^2/\text{sec}$	(92,92)	$-1/500 \text{ ft}^2/\text{sec}$	(93,93)	$-1/1500 \text{ ft}^2/\text{sec}$
(14,15)	$1 \text{ ft}^2/\text{sec}$				

B.2 Elements of the Process Noise and Measurement Noise Matrices

The process noise strength matrix \mathbf{Q} associated with the INS truth model is also partitioned into submatrices as described in Chapter III. The NON-ZERO elements of these submatrices are shown in Tables B.7 and B.8. Note that the σ^2 terms in these two tables are variable names only as defined in the Litton technical report and do not represent variance terms typically associated with σ^2 . The process noise for the GPS and RRS portions of the truth model are listed in Table B.9. Finally, the process noise values used in the filter and the measurement noise values \mathbf{R} are presented in Tables B.10 and B.11.

Table B.7. Elements of Truth Model Process Noise Submatrix Q_{11}

Element	Term	Element	Term
(4,4)	$\sigma_{\eta_{b_x}}^2$	(5,5)	$\sigma_{\eta_{b_y}}^2$
(6,6)	$\sigma_{\eta_{b_z}}^2$	(7,7)	$\sigma_{\eta_{A_x}}^2$
(8,8)	$\sigma_{\eta_{A_y}}^2$	(9,9)	$\sigma_{\eta_{A_z}}^2$

Table B.8. Elements of Truth Model Process Noise Submatrix Q_{22}

Element	Term	Element	Term	Element	Term
(11,11)	$2\beta_{\delta h_c} \sigma_{\delta h_c}^2$	(19,19)	$2\beta_{\nabla_{x_c}} \sigma_{\nabla_{x_c}}^2$	(20,20)	$2\beta_{\nabla_{y_c}} \sigma_{\nabla_{y_c}}^2$
(21,21)	$2\beta_{\nabla_{x_c}} \sigma_{\nabla_{x_c}}^2$	(22,22)	$2\beta_{\delta g_x} \sigma_{\delta g_x}^2$	(23,23)	$2\beta_{\delta g_y} \sigma_{\delta g_y}^2$
(24,24)	$2\beta_{\delta g_z} \sigma_{\delta g_z}^2$				

Table B.9. Elements of Truth Model Process Noise for GPS & RRS

(49,49)	$6.667 \times 10^{-13} \text{ ft}^2/\text{sec}$	(53,53)	$6.667 \times 10^{-13} \text{ ft}^2/\text{sec}$	(57,57)	$6.667 \times 10^{-13} \text{ ft}^2/\text{sec}$
(61,61)	$6.667 \times 10^{-13} \text{ ft}^2/\text{sec}$	(65,65)	$6.667 \times 10^{-13} \text{ ft}^2/\text{sec}$	(69,69)	$6.667 \times 10^{-13} \text{ ft}^2/\text{sec}$
(70,70)	$0.5 \text{ ft}^2/\text{sec}$	(71,71)	$0.004 \text{ ft}^2/\text{sec}$	(72,72)	$0.004 \text{ ft}^2/\text{sec}$
(77,77)	$0.5 \text{ ft}^2/\text{sec}$	(78,78)	$0.004 \text{ ft}^2/\text{sec}$	(79,79)	$0.004 \text{ ft}^2/\text{sec}$
(84,84)	$0.5 \text{ ft}^2/\text{sec}$	(85,85)	$0.004 \text{ ft}^2/\text{sec}$	(86,86)	$0.004 \text{ ft}^2/\text{sec}$
(91,91)	$0.5 \text{ ft}^2/\text{sec}$	(92,92)	$0.004 \text{ ft}^2/\text{sec}$	(93,93)	$0.004 \text{ ft}^2/\text{sec}$

Table B.10. Filter Process Noise Q

Element	Term	Element	Term
(1,1)	$0.1 \times 10^{-13} \text{ \& } 1.0 \times 10^{-13} \text{ rad}^2/\text{sec}$	(2,2)	$0.58 \times 10^{-13} \text{ \& } 1.0 \times 10^{-13} \text{ rad}^2/\text{sec}$
(3,3)	$0.0 \text{ rad}^2/\text{sec}$	(4,4)	$100 \text{ rad}^2/\text{sec}$
(5,5)	$500 \text{ rad}^2/\text{sec}$	(6,6)	$45 \text{ rad}^2/\text{sec}$
(7,7)	$800 \text{ ft}^2/\text{sec}^3$	(8,8)	$400 \text{ ft}^2/\text{sec}^3$
(9,9)	$8000 \text{ ft}^2/\text{sec}^3$	(10,10)	$20 \text{ ft}^2/\text{sec}^2$
(11,11)	$400 \text{ ft}^2/\text{sec}^2$	(12,12)	$17 \text{ \& } 40 \text{ ft}^2/\text{sec}^2$
(13,13)	$0.0 \text{ ft}^2/\text{sec}^3$	(14,14)	$14 \text{ \& } 40 \text{ ft}^2/\text{sec}^2$
(15,15)	$0.5 \times 10^{-13} \text{ ft}^2/\text{sec}^2$		

Table B.11. Truth and Filter Measurement Noises R

Measurement	Truth Noise	Filter Noise
Baro Altimeter	2500 ft^2	2500 ft^2
Doppler	$0.02 \text{ ft}^2/\text{sec}^2$	$0.02 \text{ ft}^2/\text{sec}^2$
Transponders	4 ft^2	100 ft^2
Satellites	2 ft^2	75 ft^2

Appendix C. *Miscellaneous Plots*

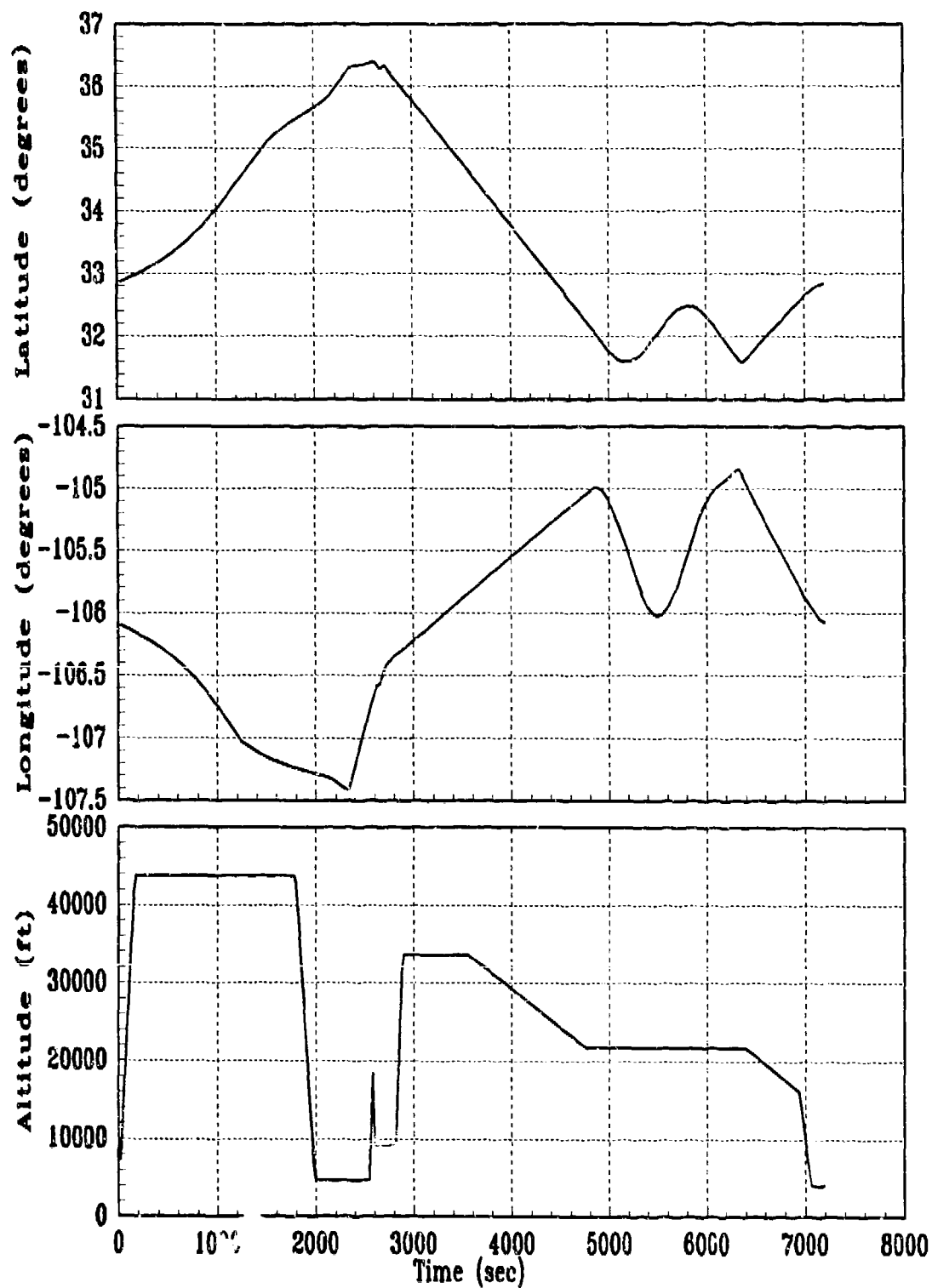


Figure C.1. Two Hour Flight Profile

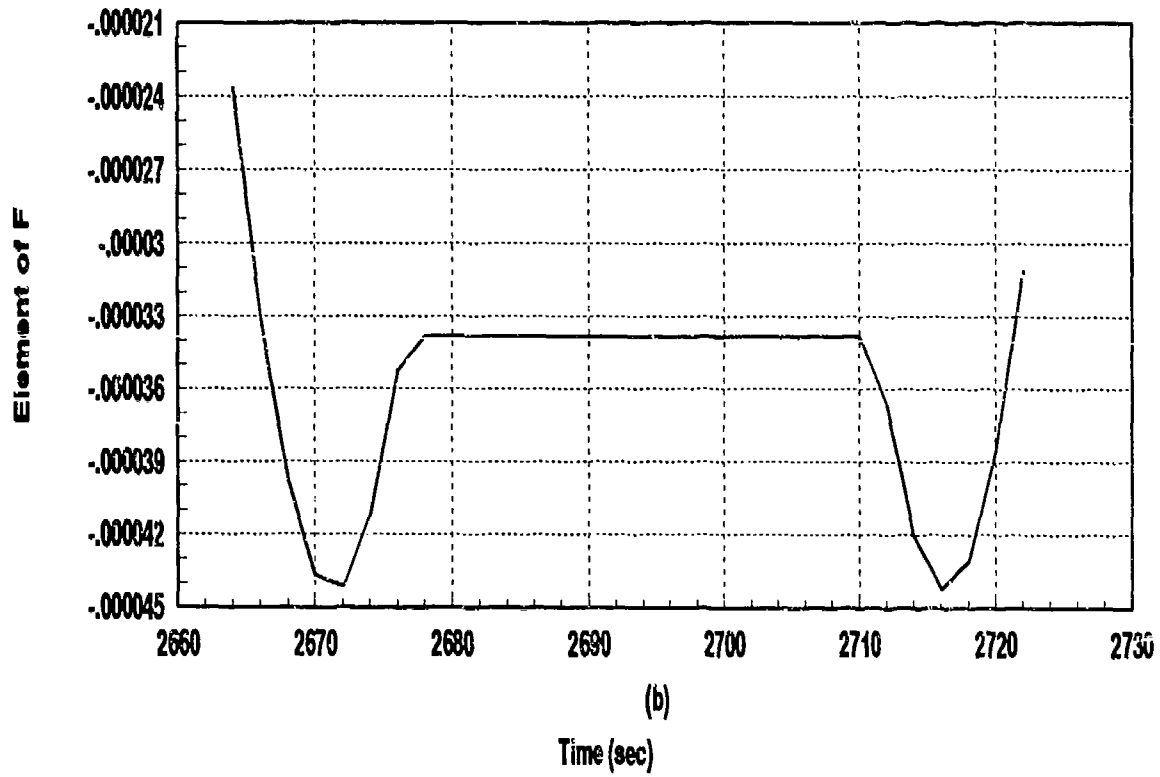
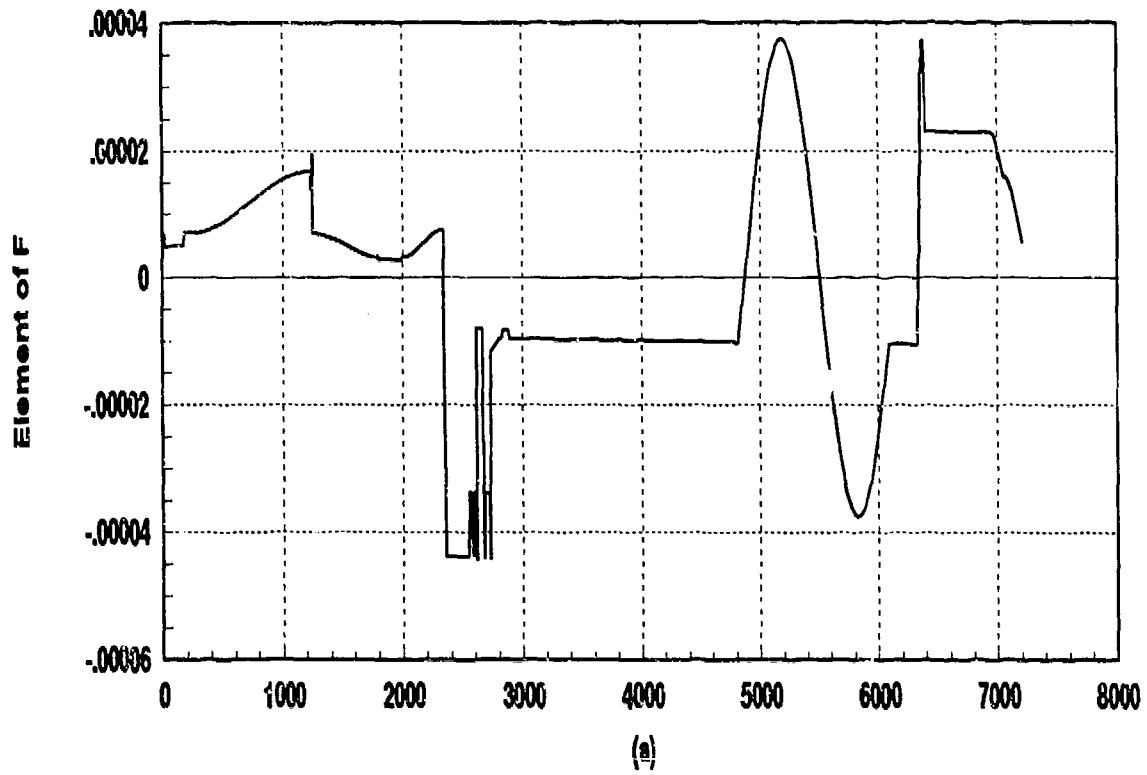


Figure C.2. Element of the State Dynamics Matrix F

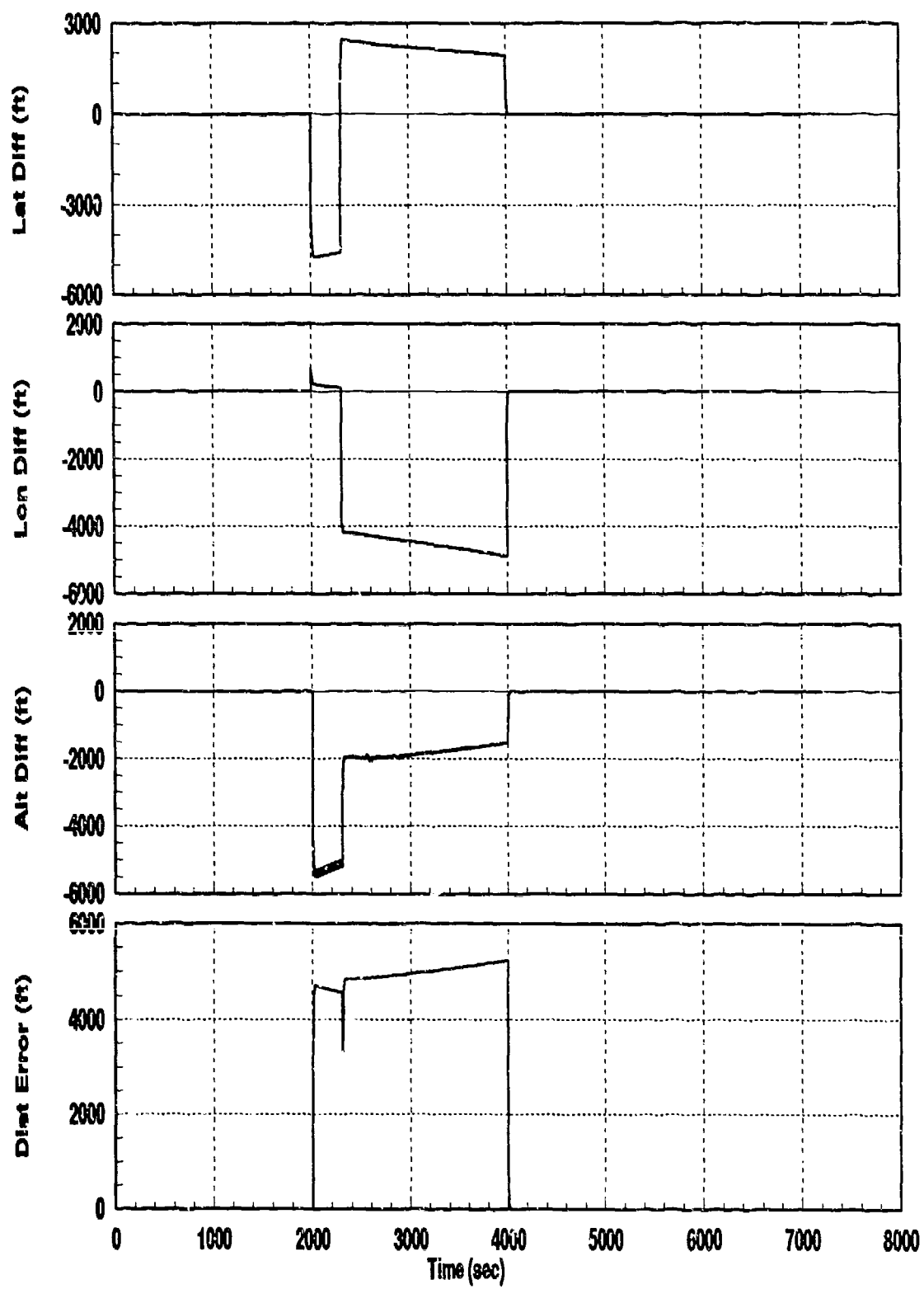


Figure C.3. Spoofing Induced Errors

Appendix D. Baseline Filter Plots

All the state plots contained in this appendix contain five traces. The innermost trace (---) on each data plot is the *mean error* time history for the applicable state. Mean Error is defined as being the difference between the filter's estimate of the state and the true state, averaged over the number of Monte Carlo runs performed. The equation describing this relationship is defined by (13, 27):

$$\hat{M}_e(t_i) = \frac{1}{N} \sum_{j=1}^N e_j(t_i) = \frac{1}{N} \sum_{j=1}^N \{\hat{x}_j(t_i) - x_{true_j}(t_i)\} \quad (D.1)$$

where $\hat{x}_j(t_i)$ is the filter-computed estimate of a given state and $x_{true_j}(t_i)$ is the *truth* model value of the same state, at time t_i , for run j , and N is the number of time histories in the simulation (10 in this thesis).

In addition to the center trace, two more *pairs* of traces are plotted and labeled *Mean+-Sigma*. The first pair (represented by ...) is symmetrically displaced about the mean and as a result follows the "undulations" of the $\hat{M}_e(t_i)$. The locus of these traces is calculated from $\hat{M}_e(t_i) \pm \sqrt{P_e(t_i)}$, where $P_e(t_i)$ is the *true* error variance at time t_i . The true standard deviation is calculated from (13, 27):

$$\sigma_{true}(t_i) = \sqrt{P_e(t_i)} = \sqrt{\frac{1}{N-1} \sum_{j=1}^N \left(\hat{x}_j(t_i) - \bar{\hat{x}}(t_i) \right)^2} \quad (D.2)$$

where N is the number of runs in the Monte Carlo simulation (10 in this thesis), and $\bar{\hat{x}}^2(t_i)$ is the square of the mean of a given state at each time of interest (such as measurement times).

The last pair of traces (—) represent the filter computed $\pm \sigma_{filter}$ values for the same states and are symmetrically displaced about zero because the filter "believes" that it is producing zero-mean errors (15, 27). These quantities are propagated and updated in the MSOFE (22, 27) software using the covariance propagation equation shown in Chapter II. These traces represent the filter's estimate of its own error.

Similar statistics are computed for the measurement residuals with the residual covariance defined in Chapter II and labeled *Filter+-Sigma*. Finally, baseline plots for the GLR and chi-square algorithms are shown, with three window sizes presented for the chi-square test.

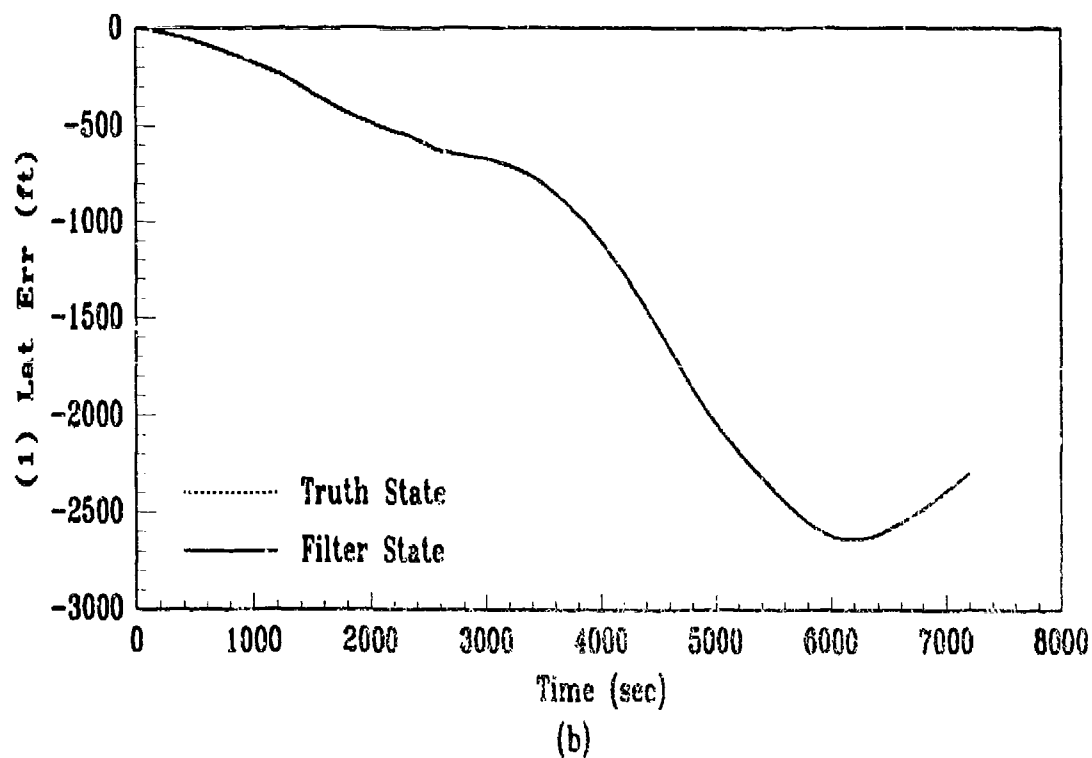
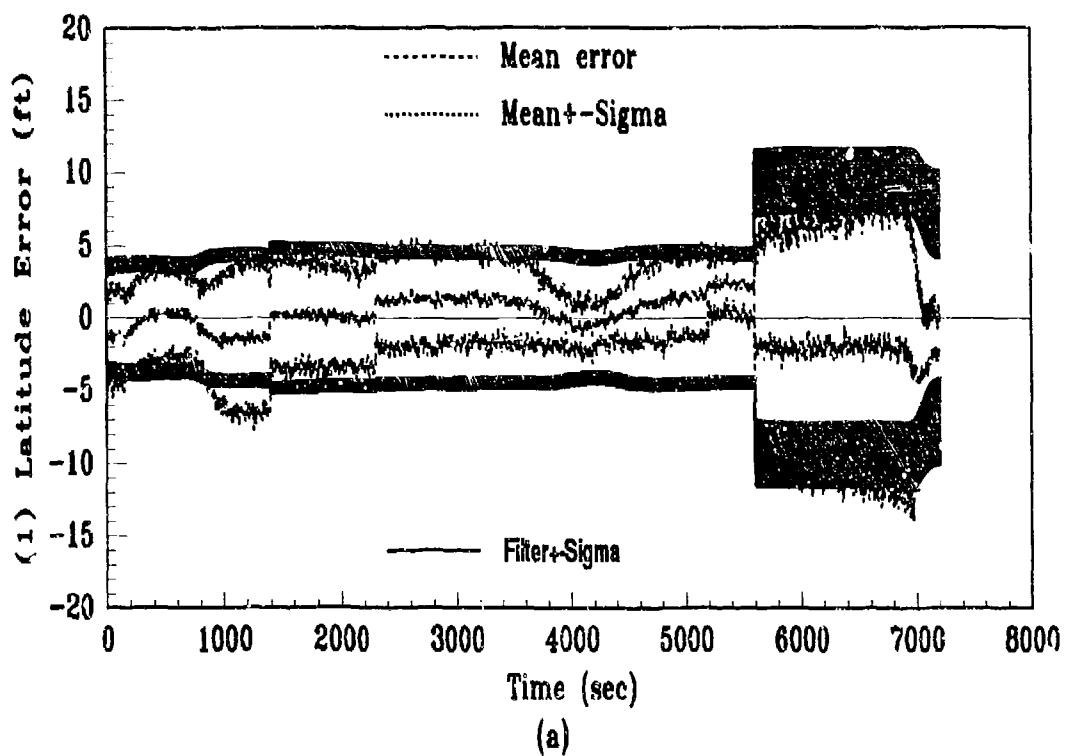


Figure D.1. State Plots: Baseline

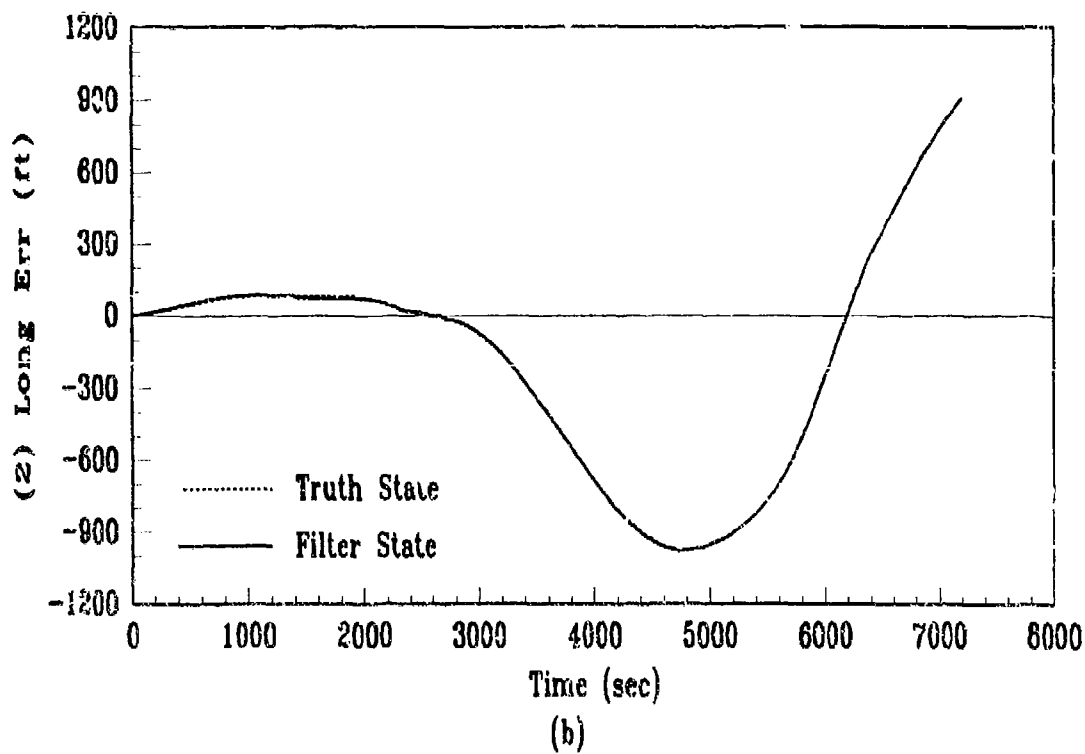
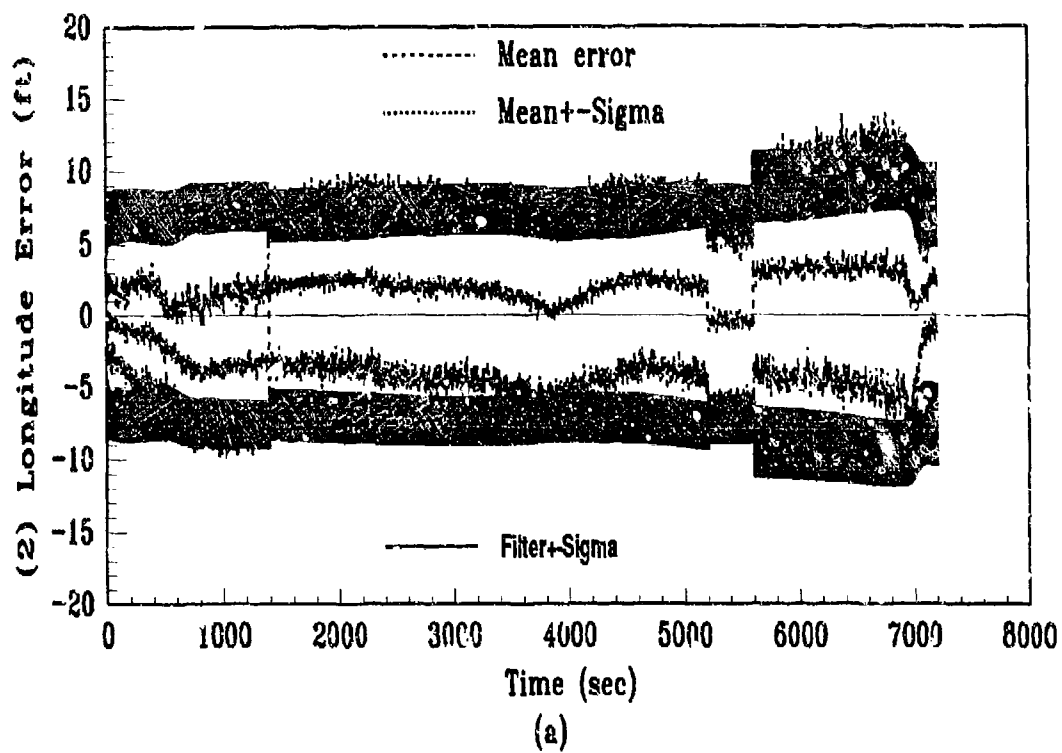


Figure D.2. State Plots: Baseline

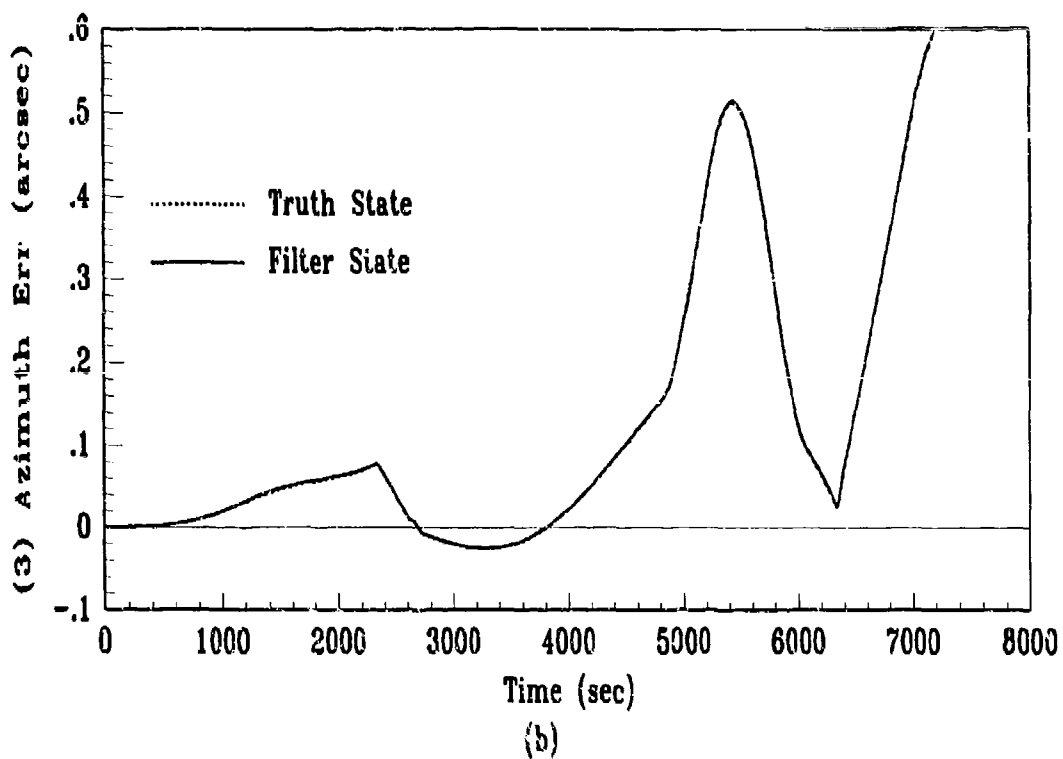
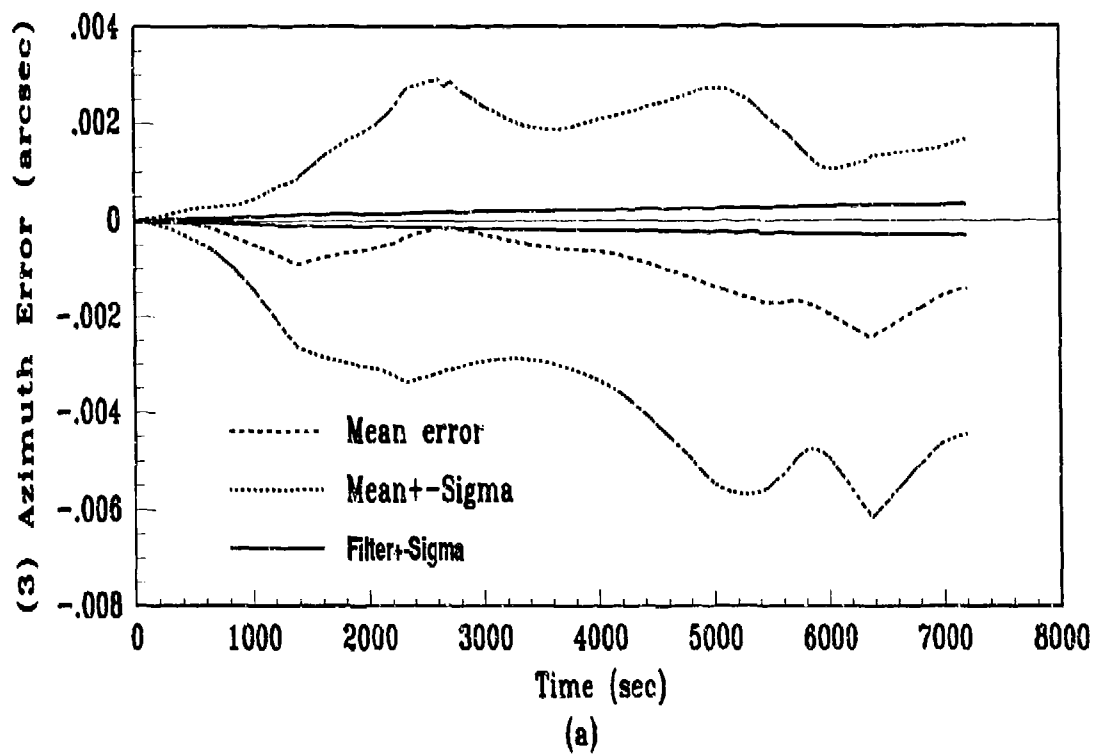


Figure D.3. State Plots: Baseline

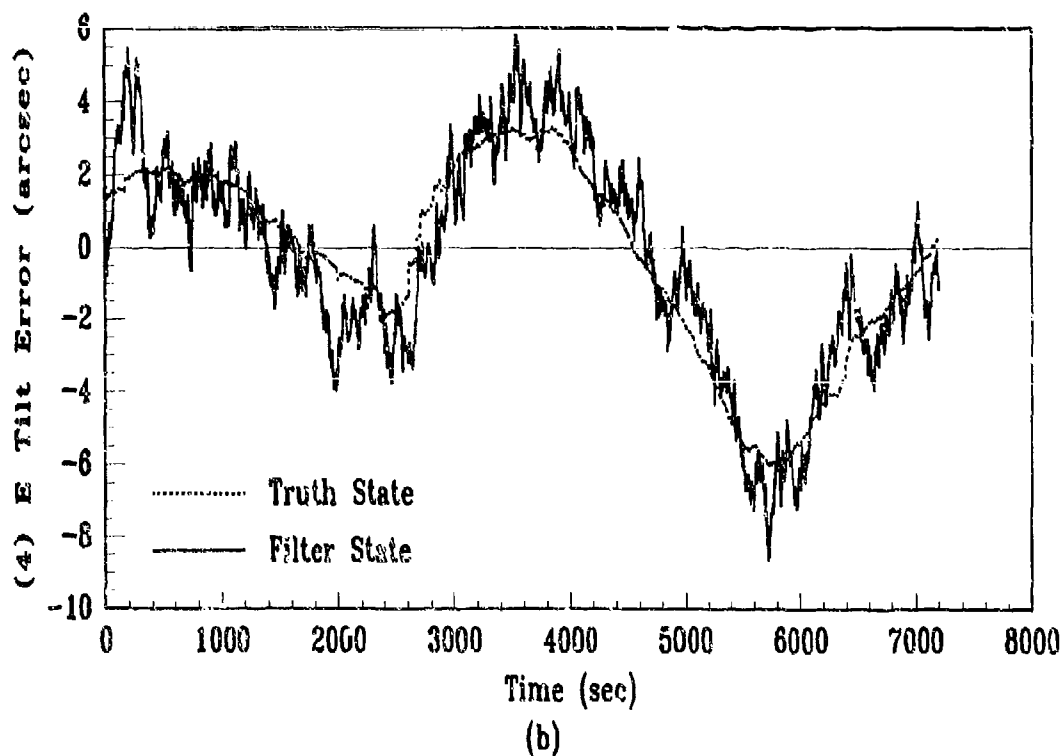
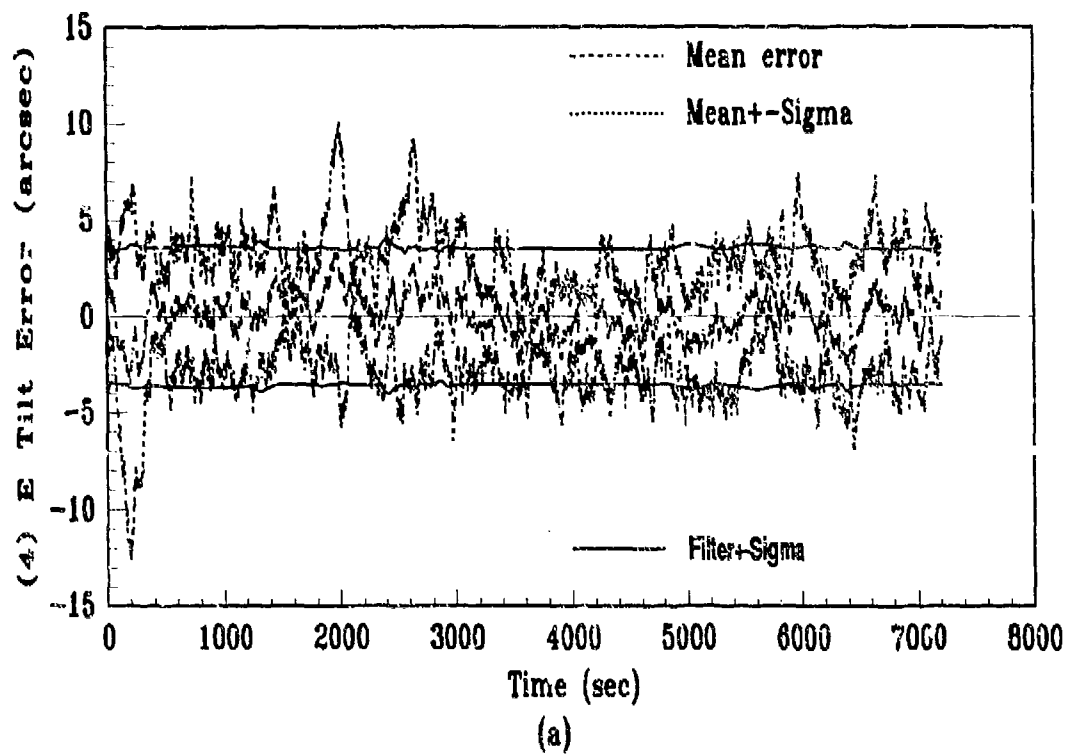


Figure D.4. State Plots: Baseline

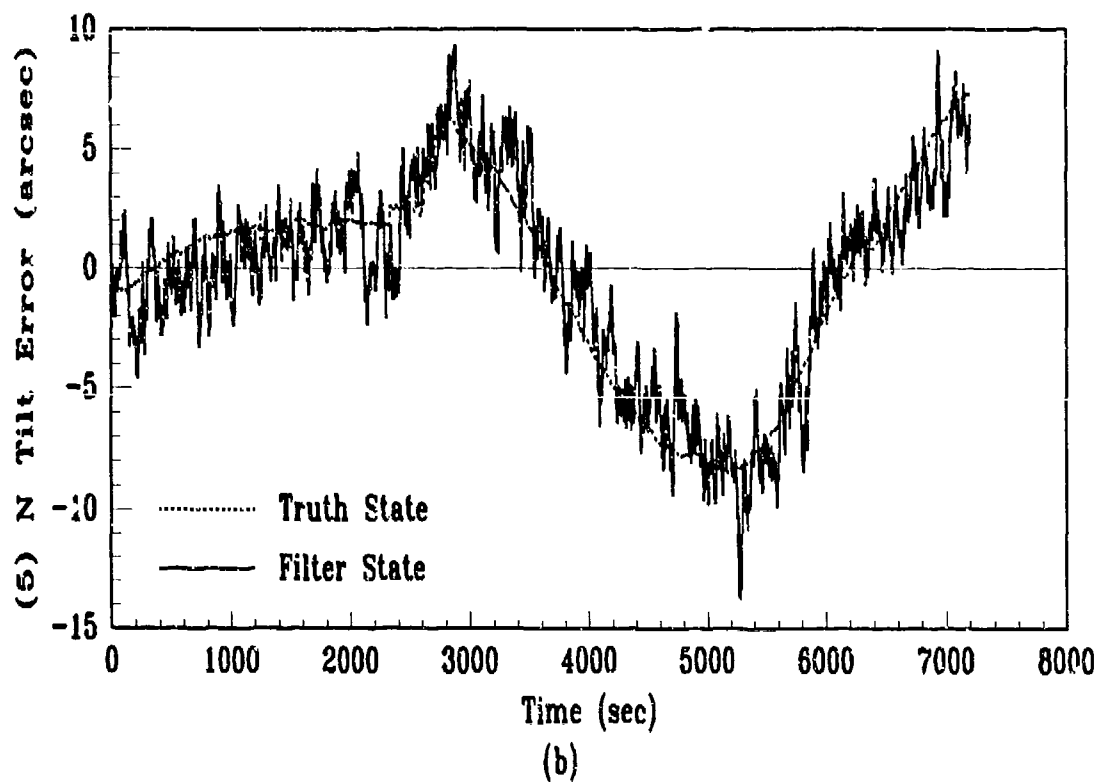
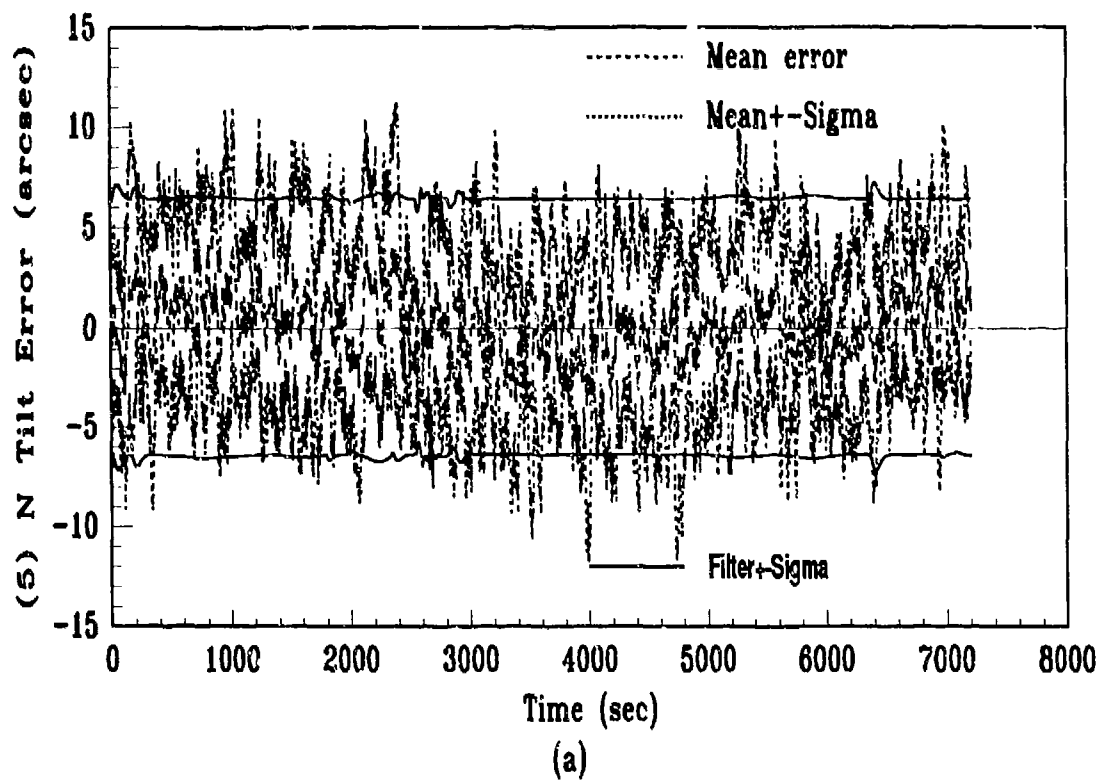


Figure D.5. State Plots: Baseline

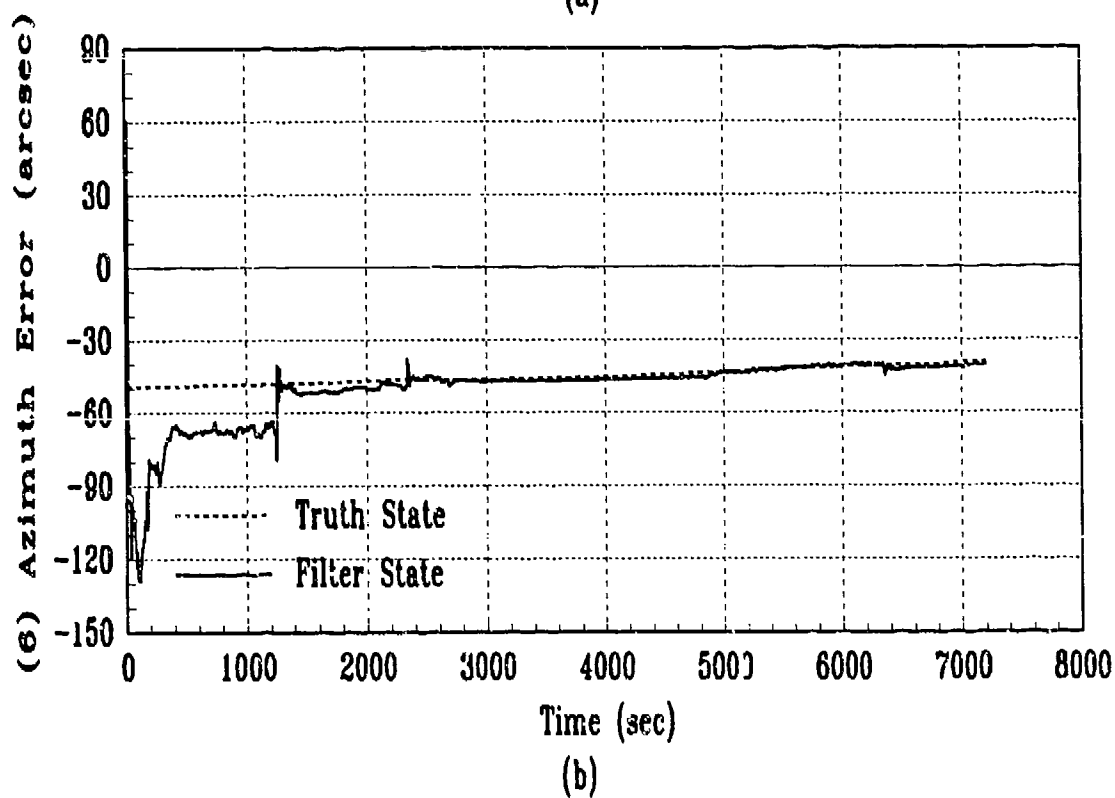
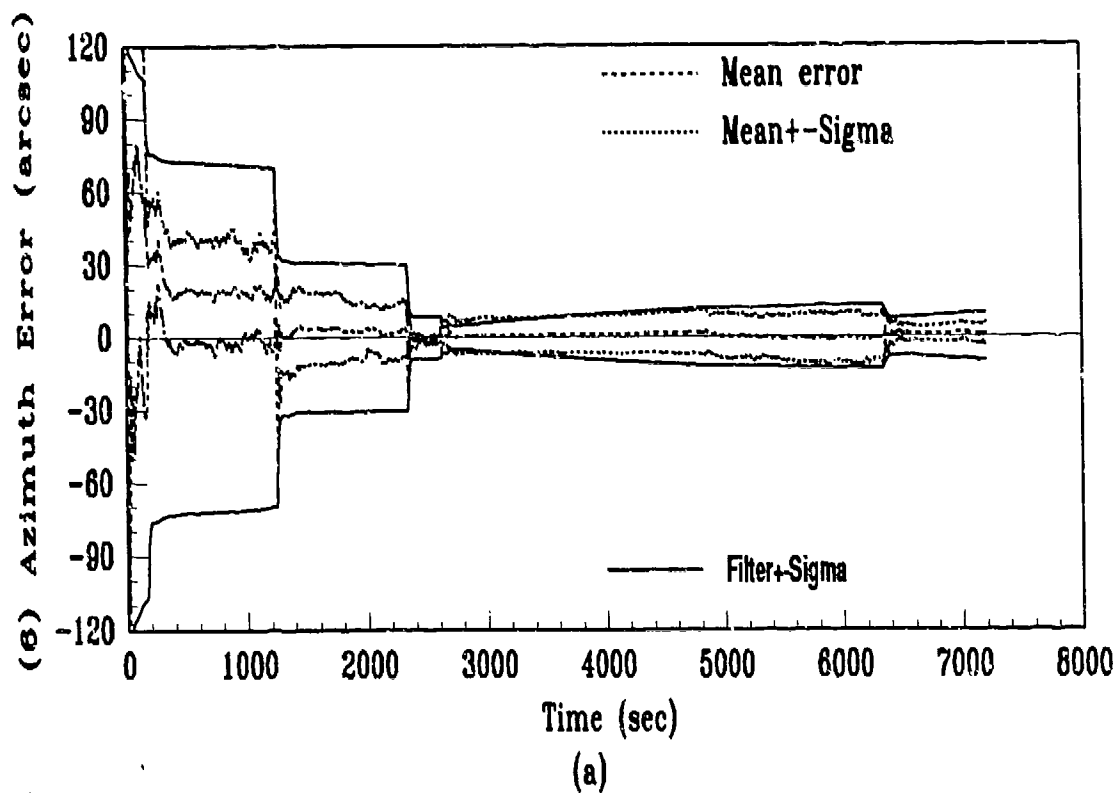


Figure D.6. State Plots: Baseline

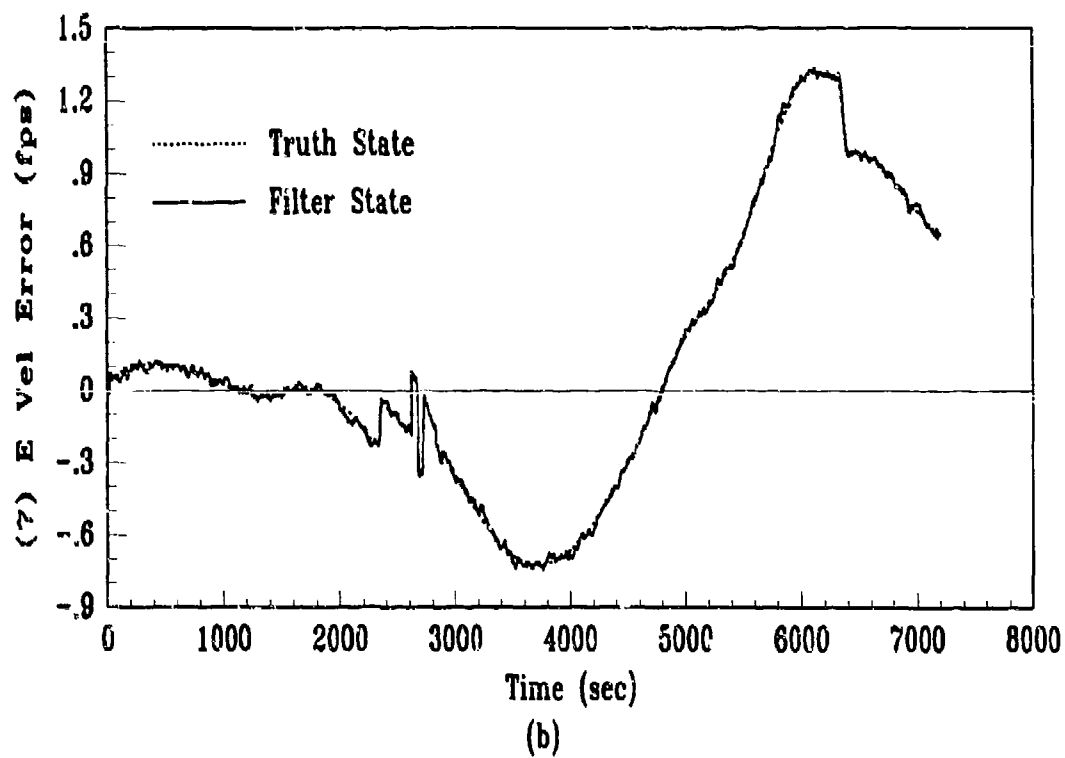
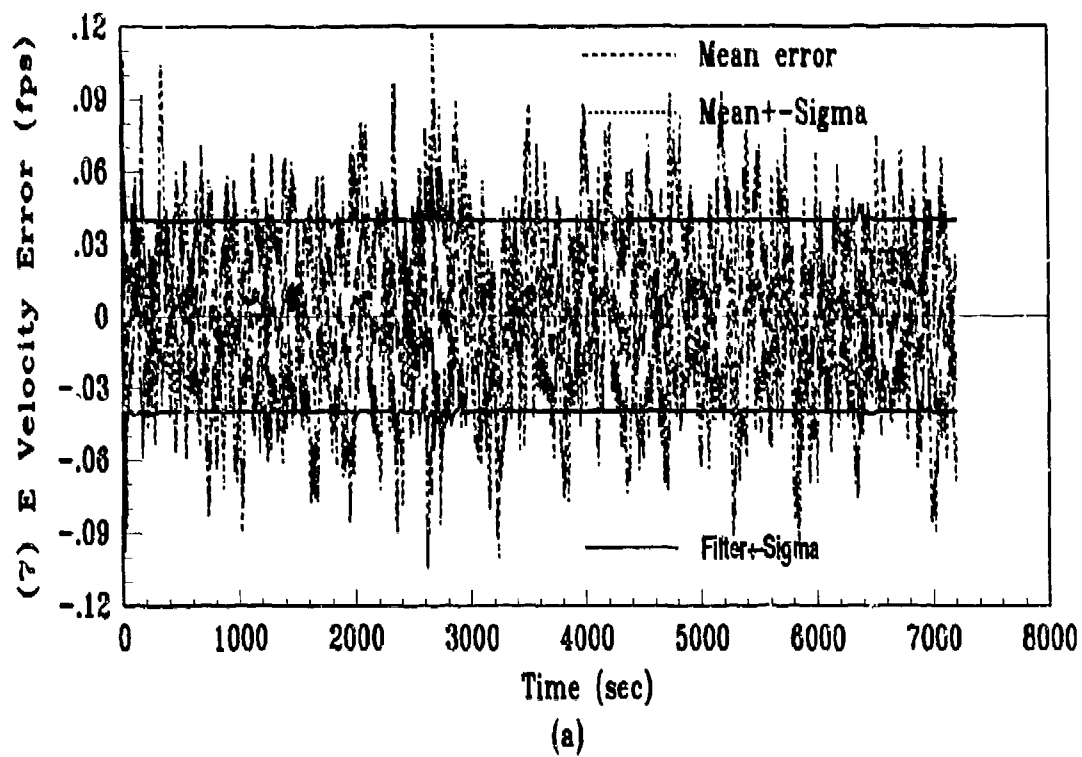


Figure D.7. State Plots: Baseline

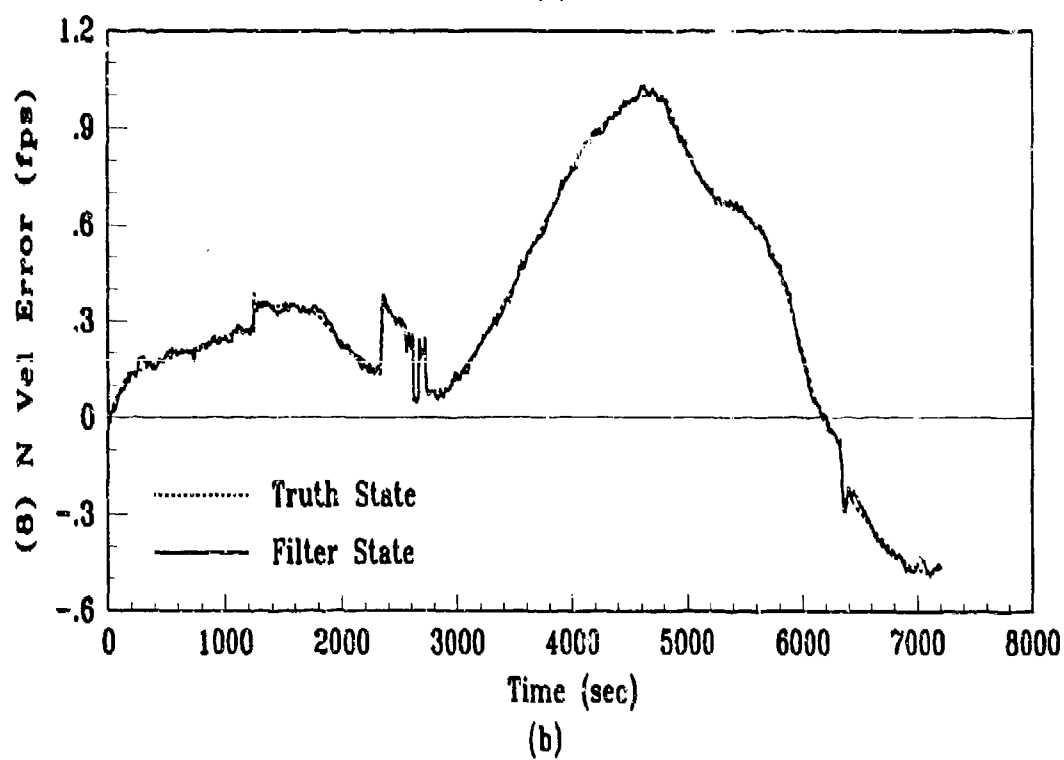
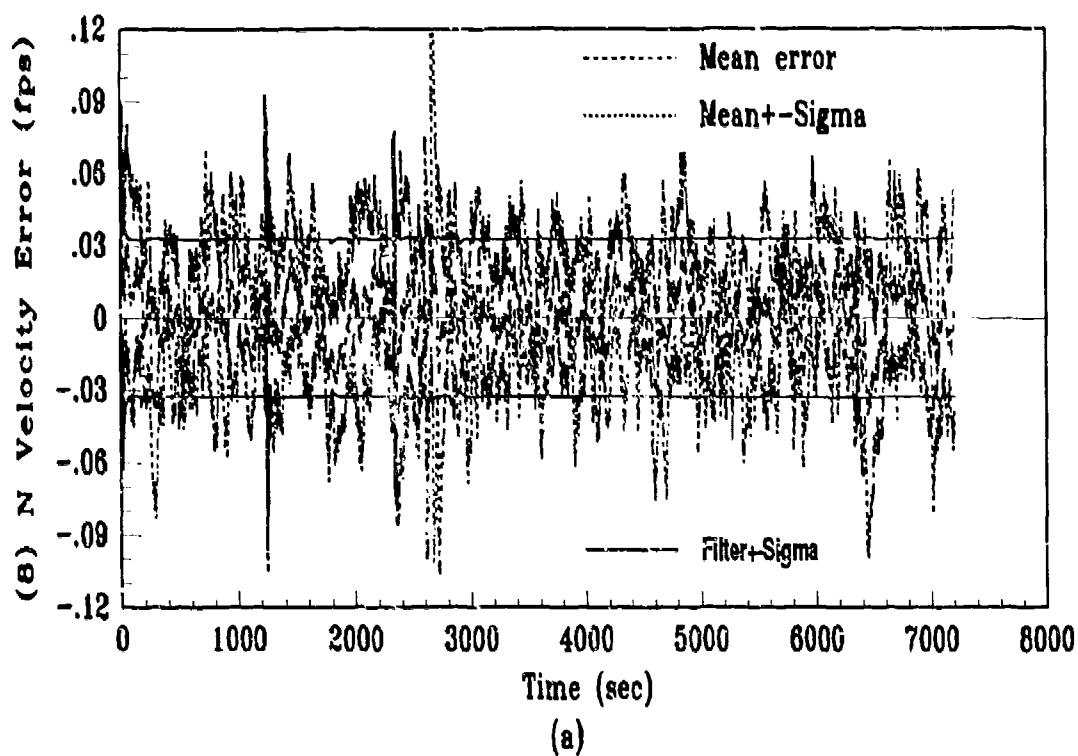


Figure D.8. State Plots: Baseline

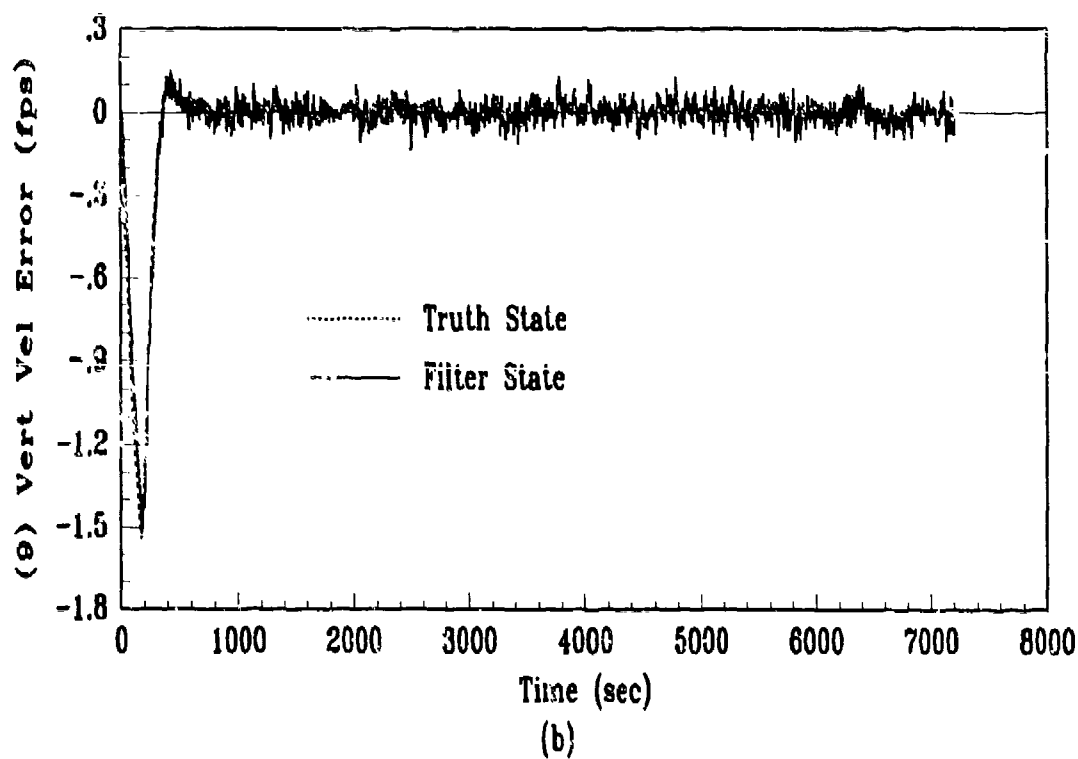
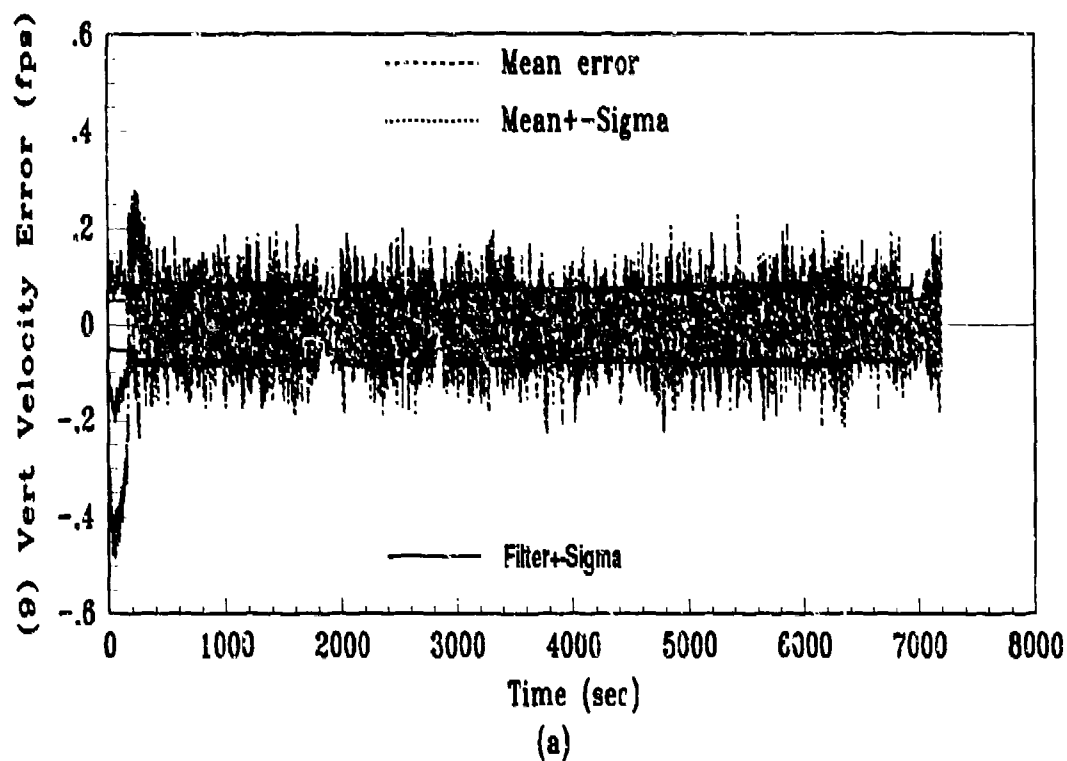


Figure D.9. State Plots: Baseline

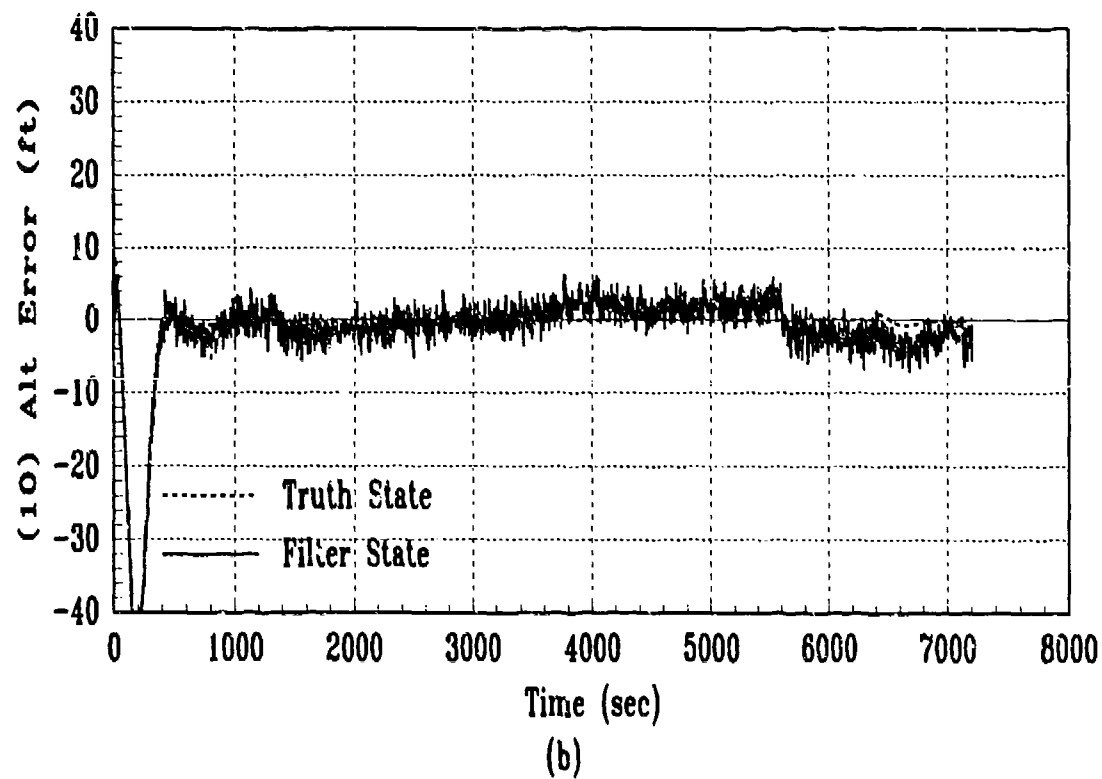
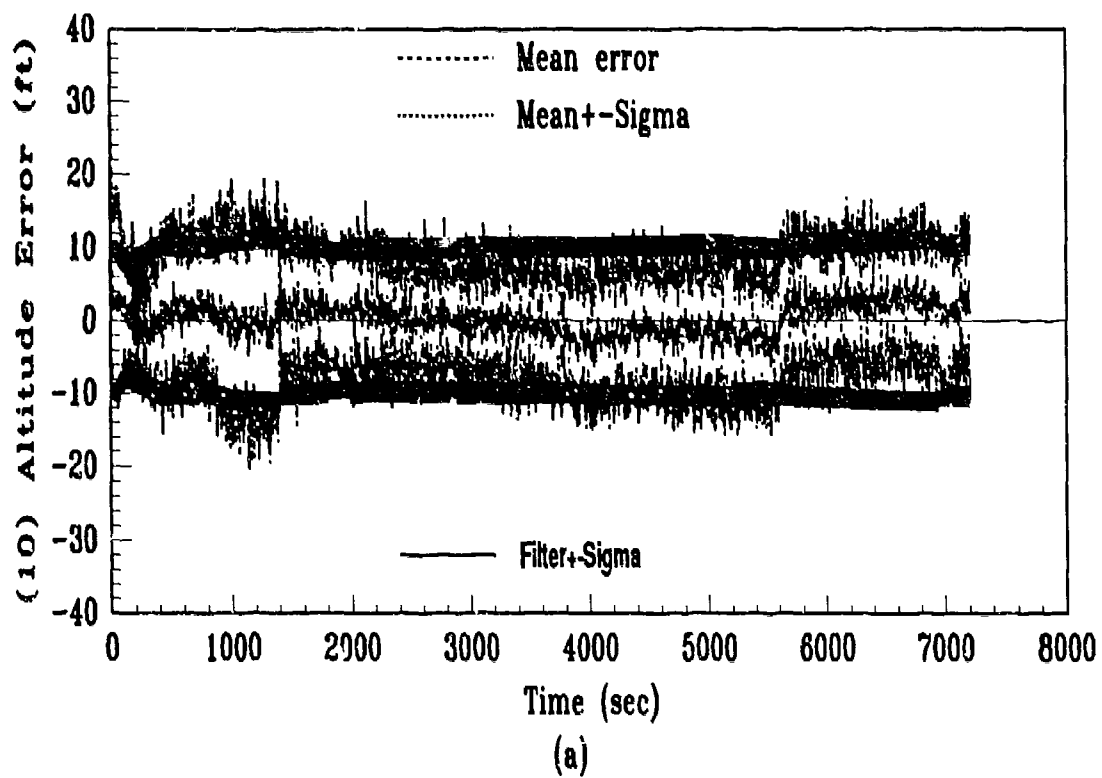


Figure D.10. State Plots: Baseline

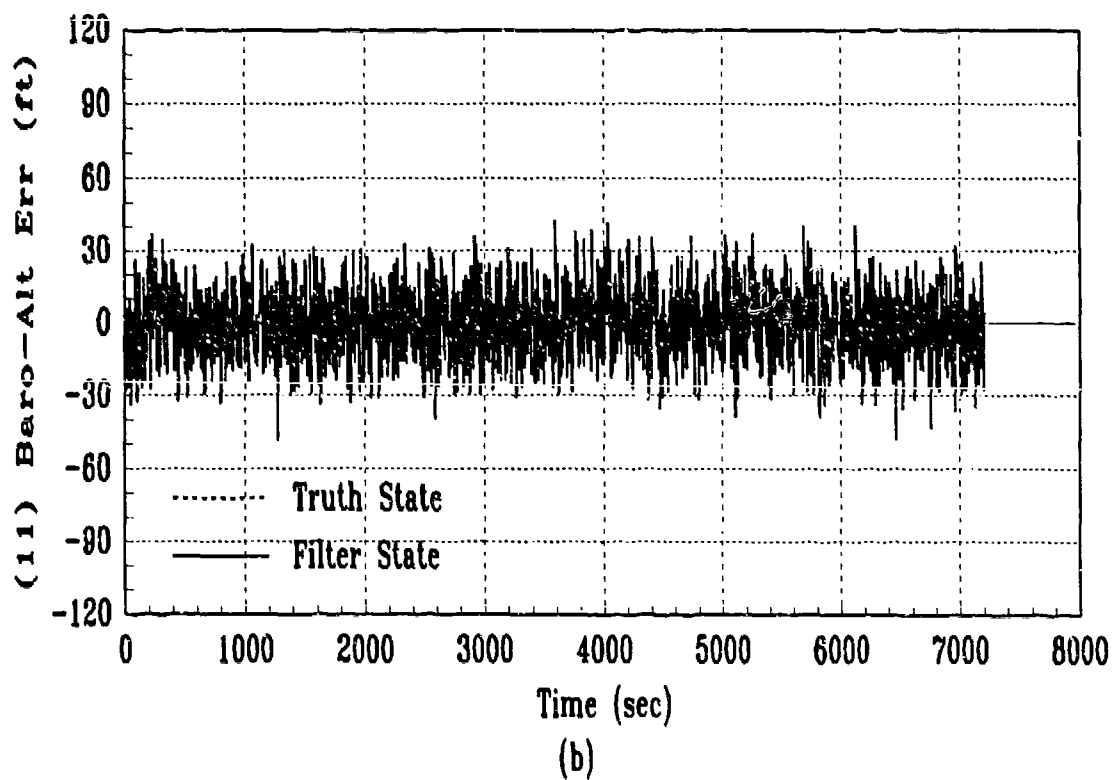
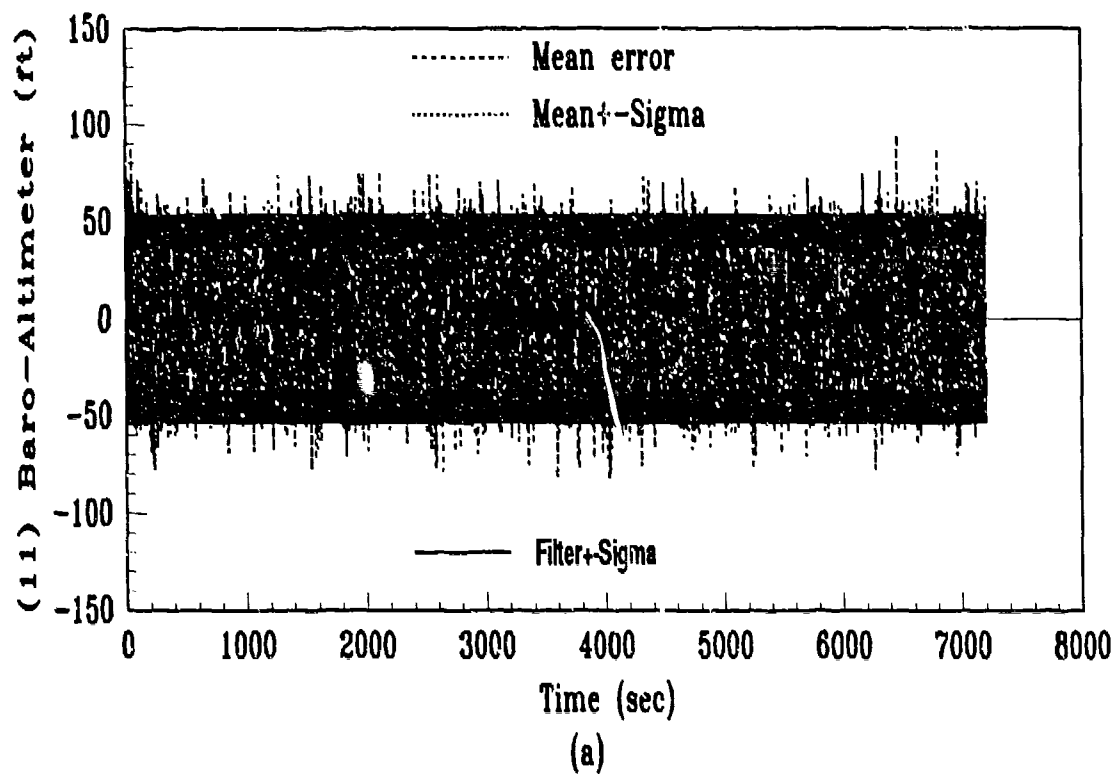


Figure D.11. State Plots: Baseline

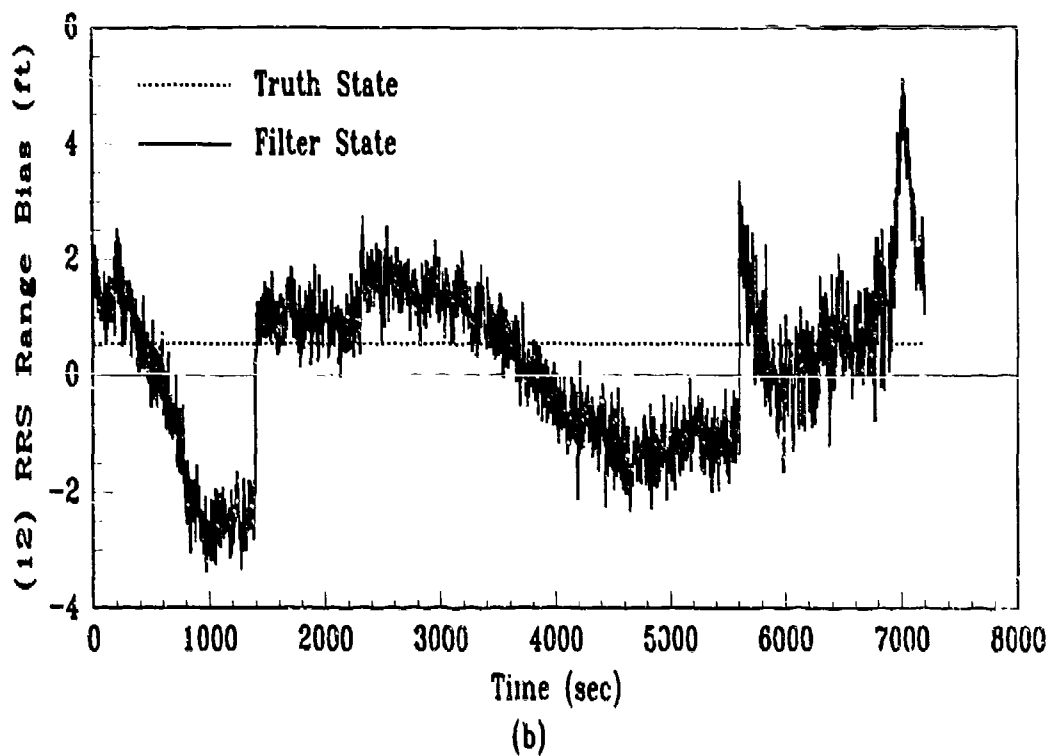
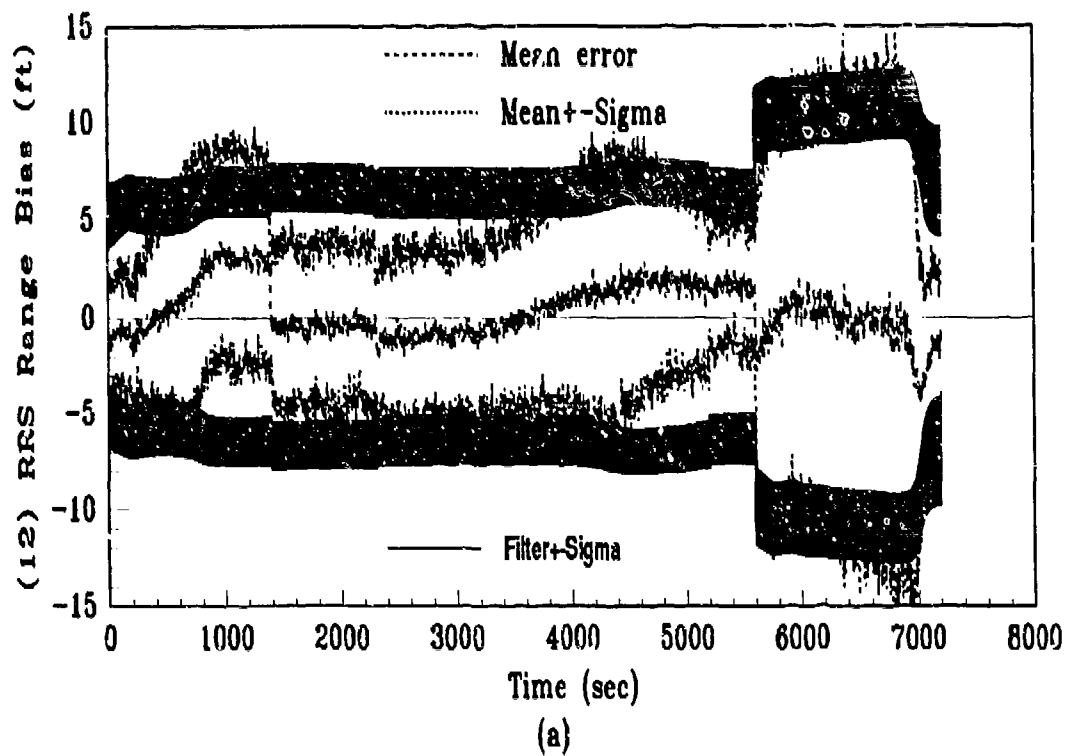


Figure D.12. State Plots: Baseline

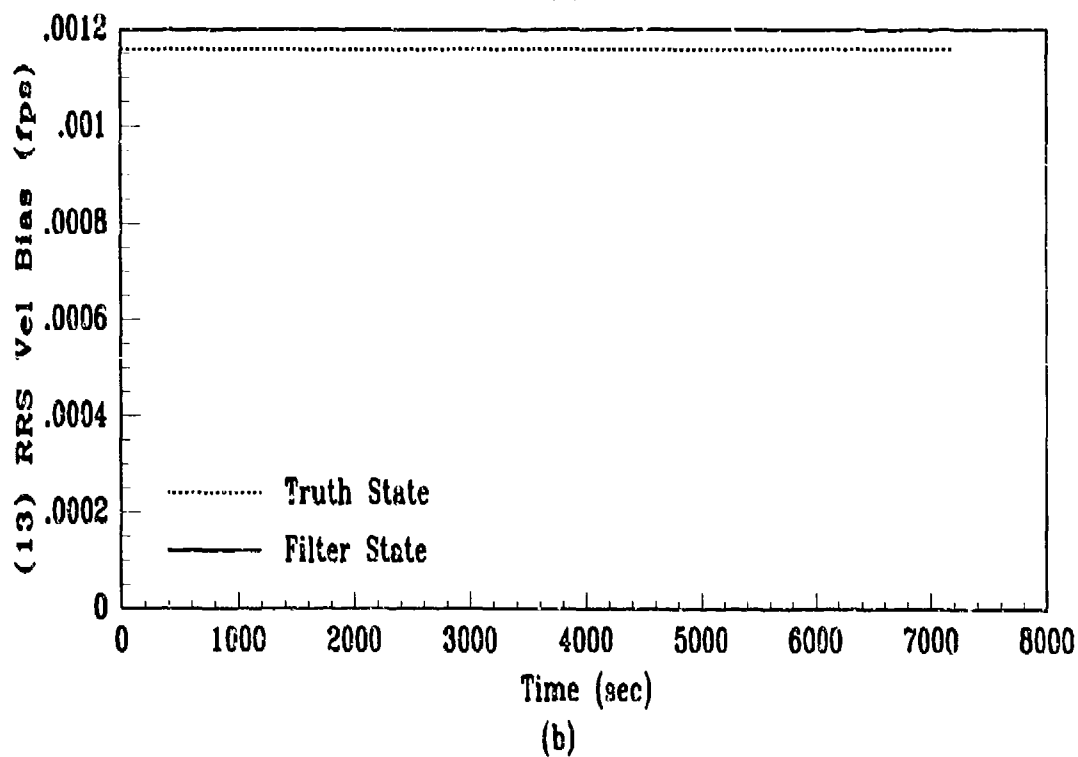
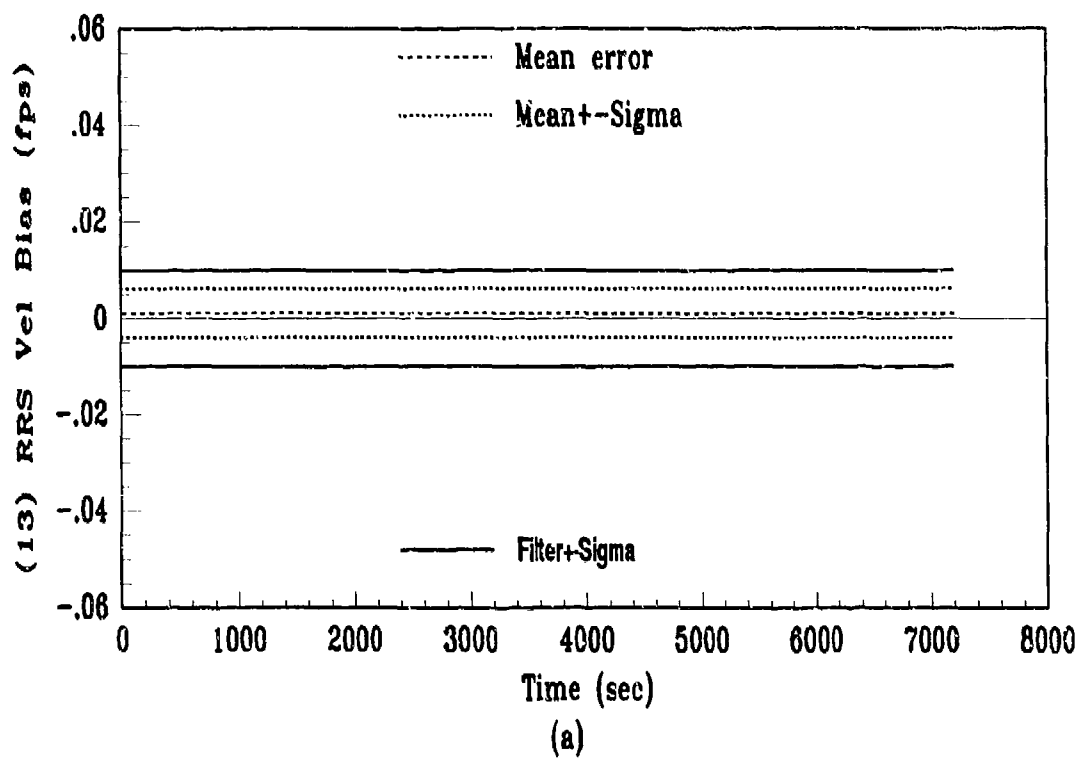


Figure D.13. State Plots: Baseline

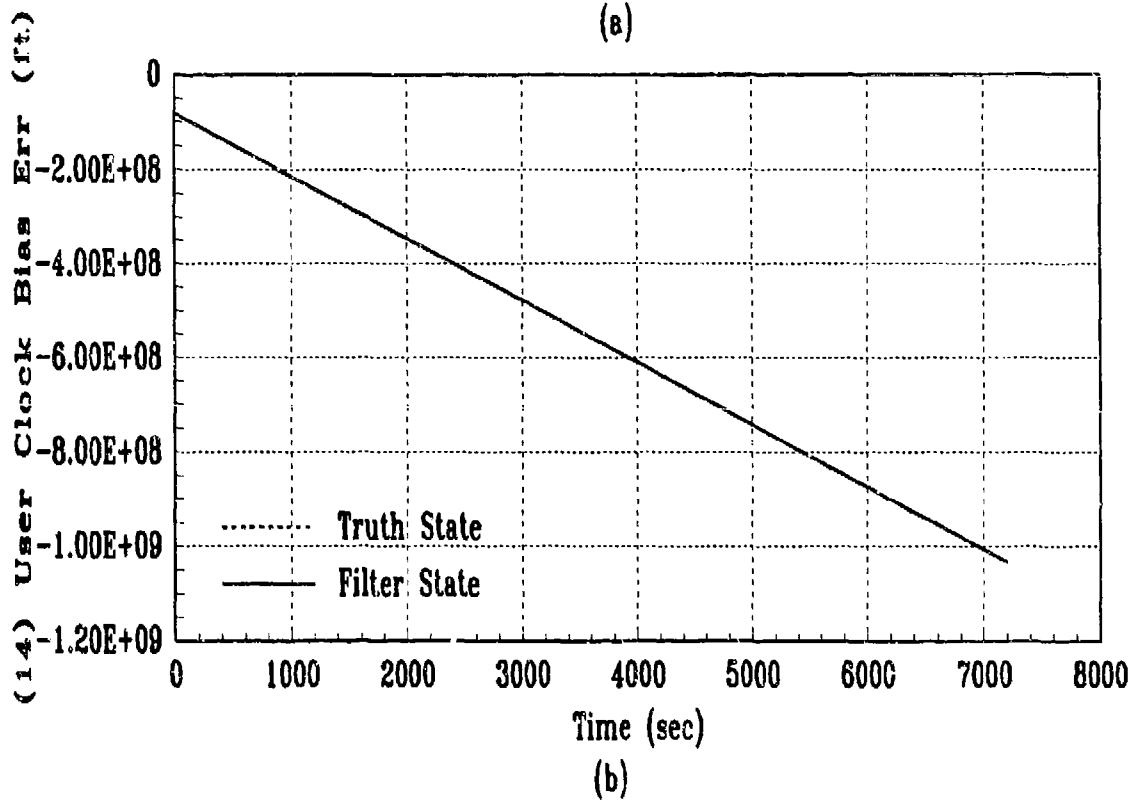
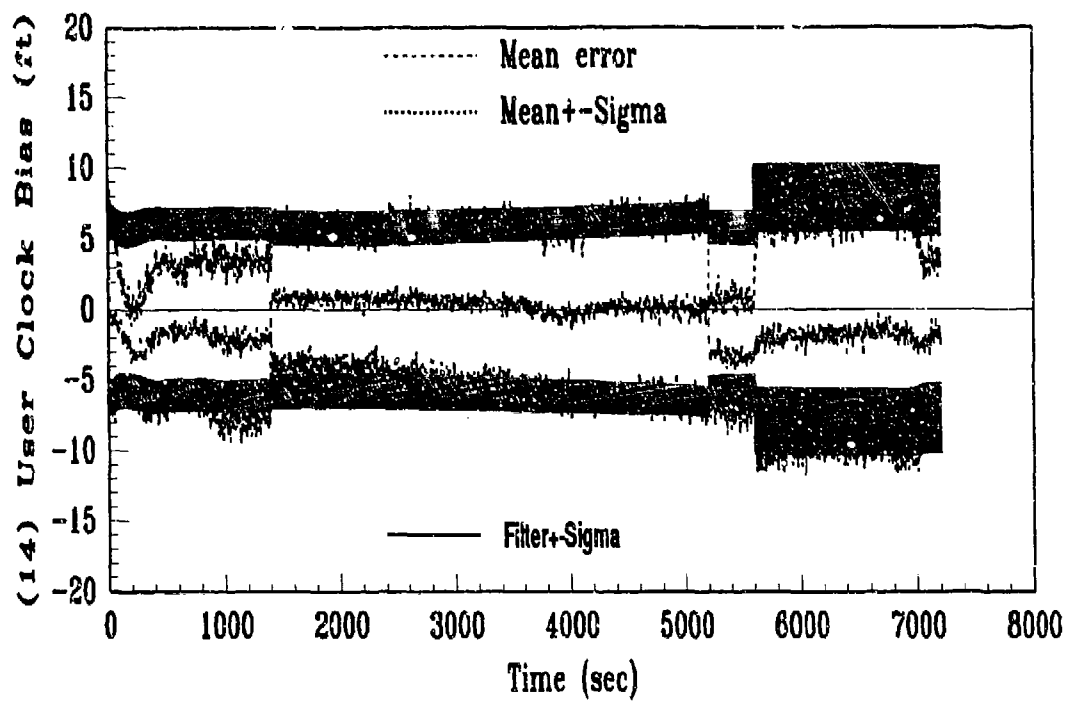


Figure D.14. State Plots: Baseline

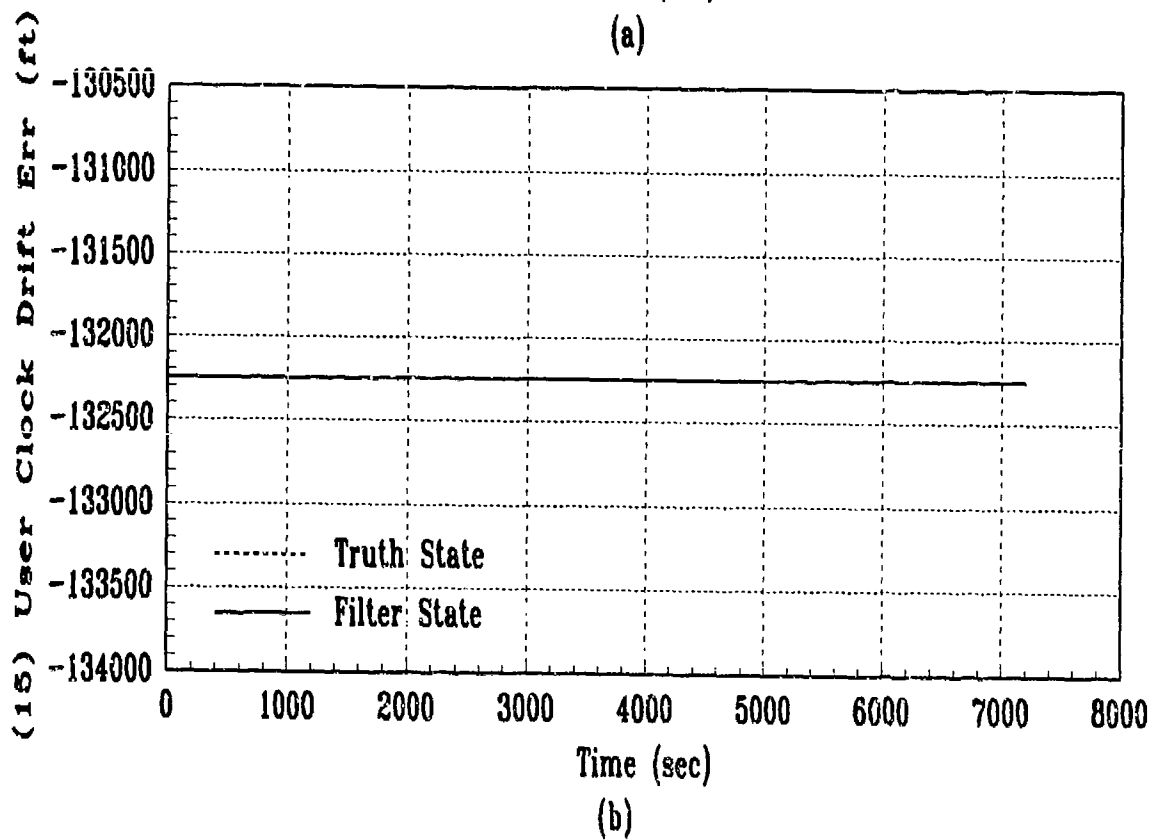
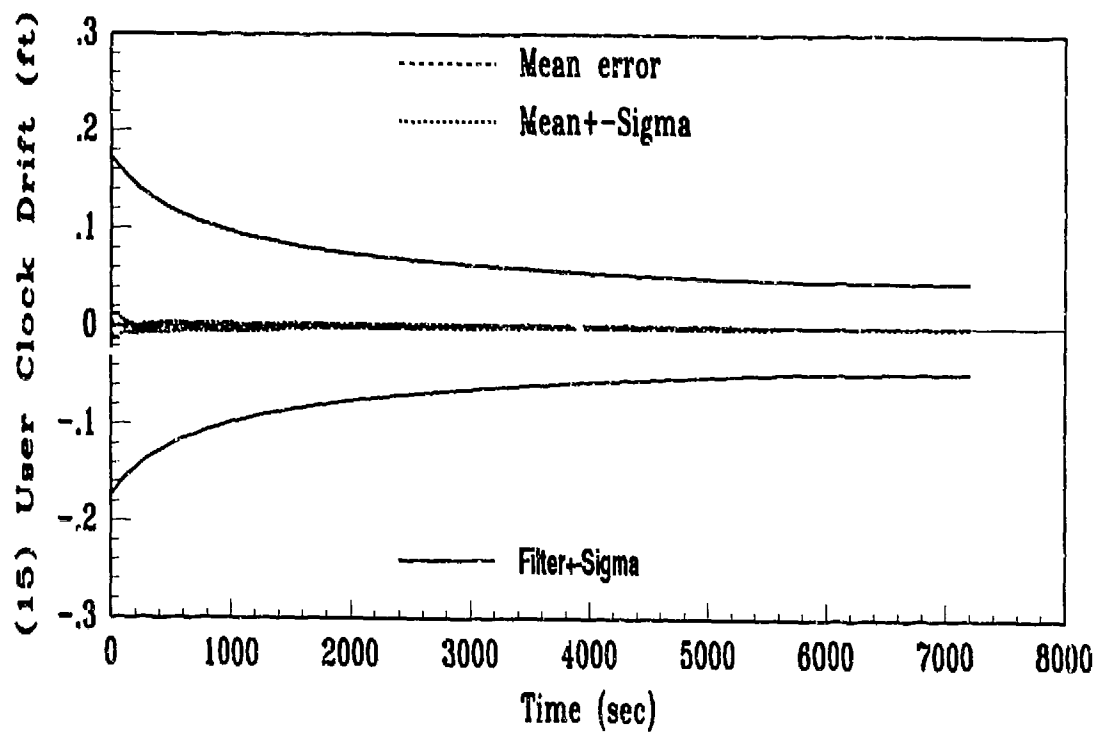


Figure D.15. State Plots: Baseline

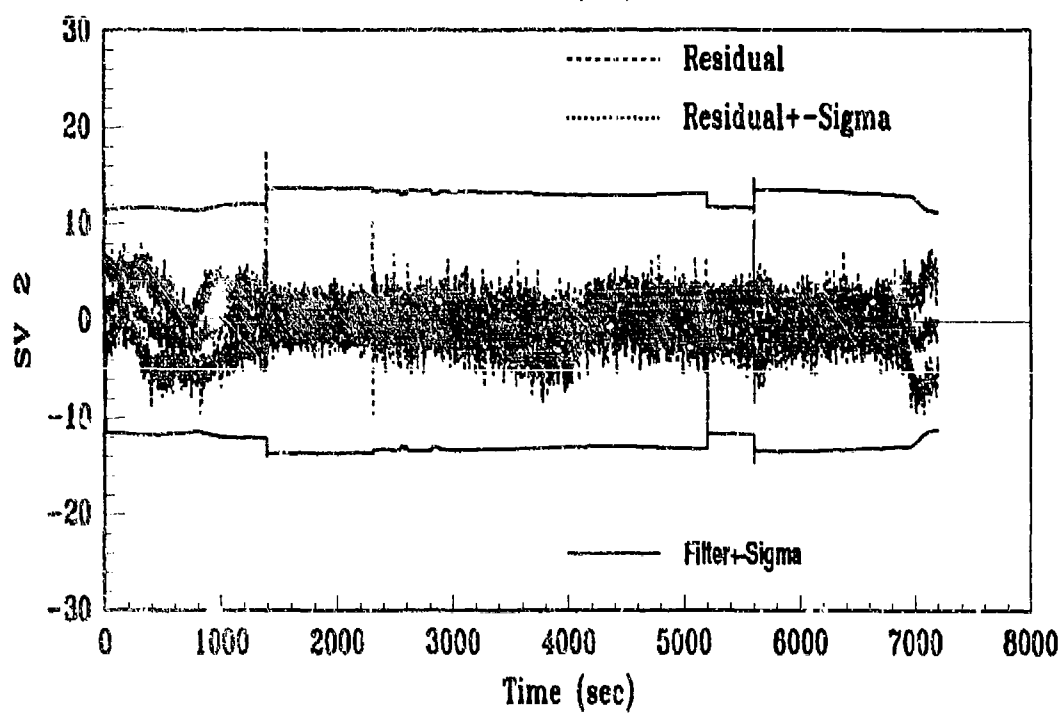
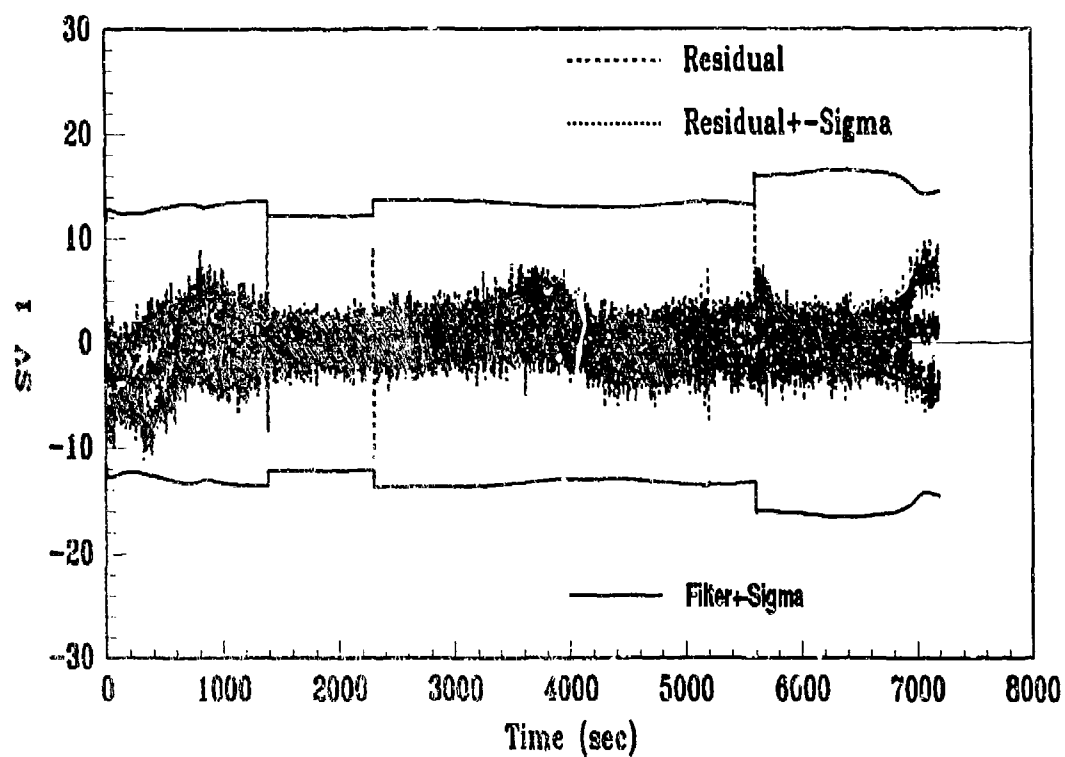


Figure D.16. Residual Plots: Baseline

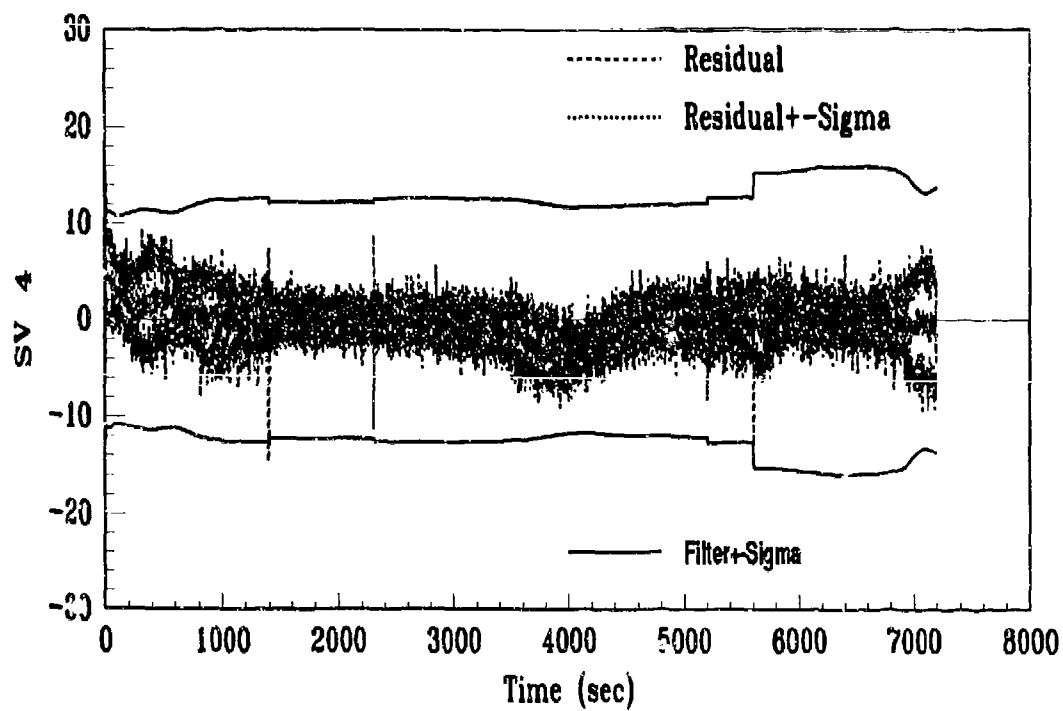
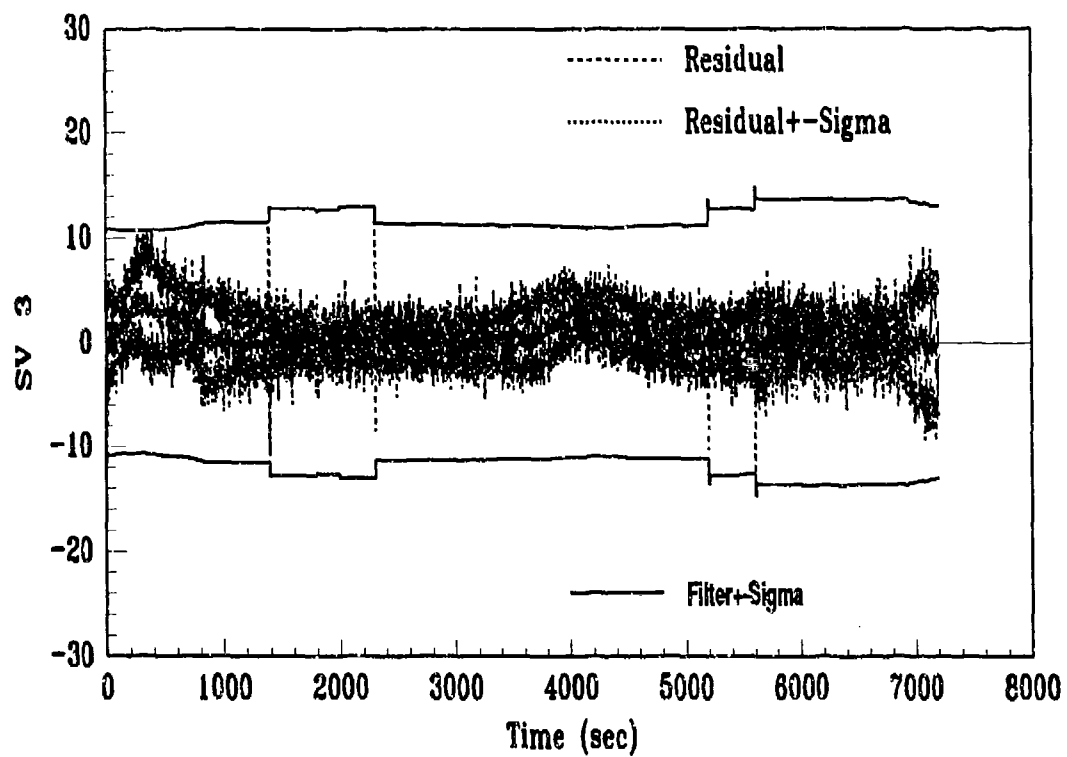


Figure D.17. Residual Plots: Baseline

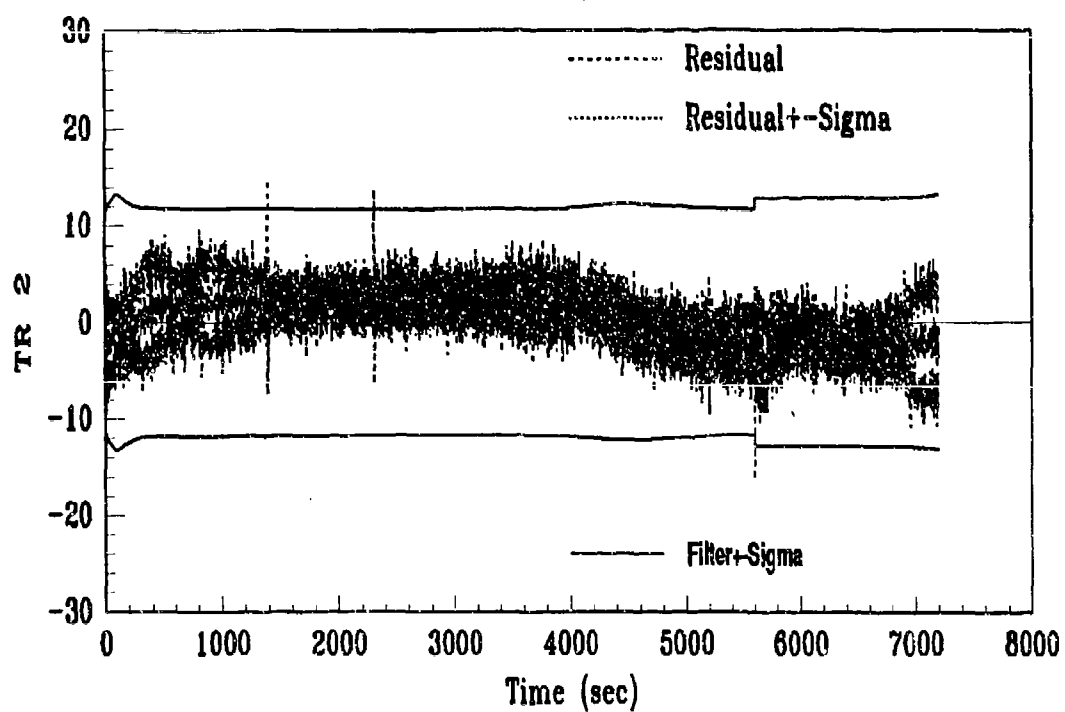
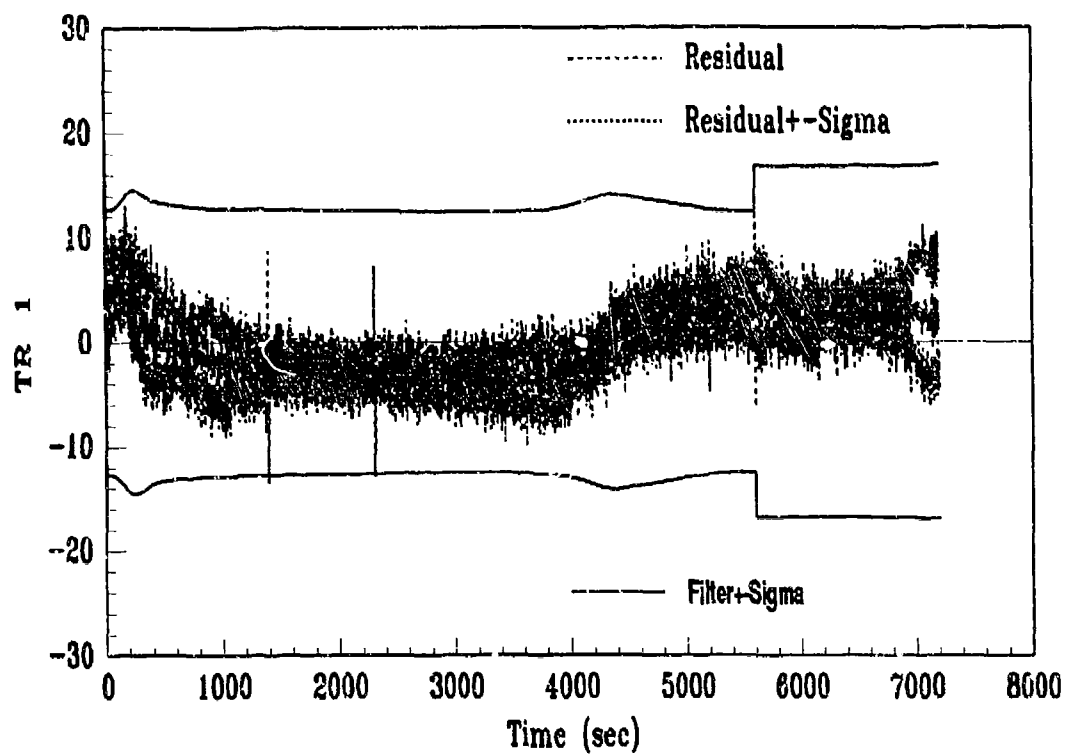


Figure D.18. Residual Plots: Baseline

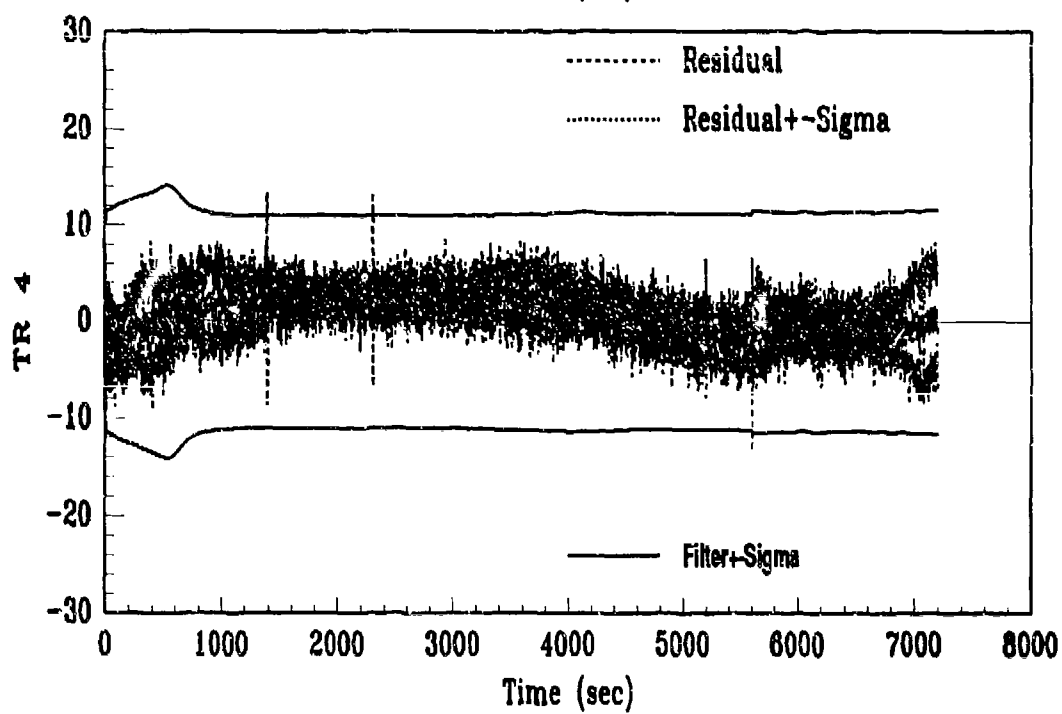
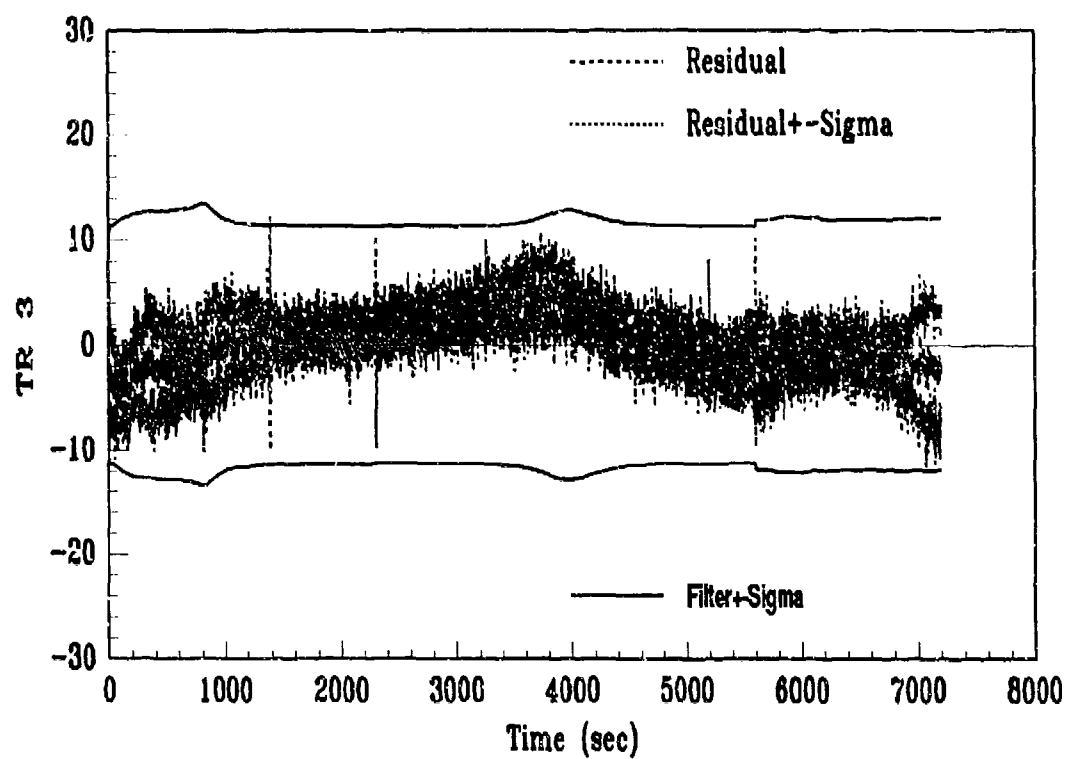


Figure D.19. Residual Plots: Baseline

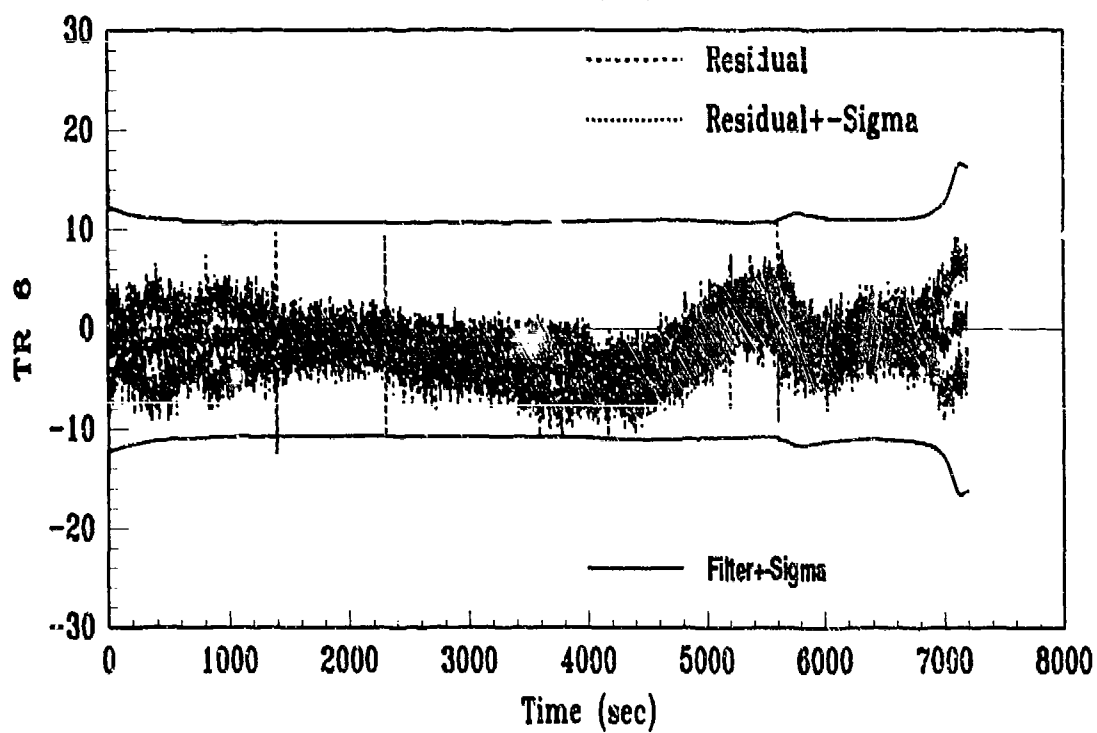
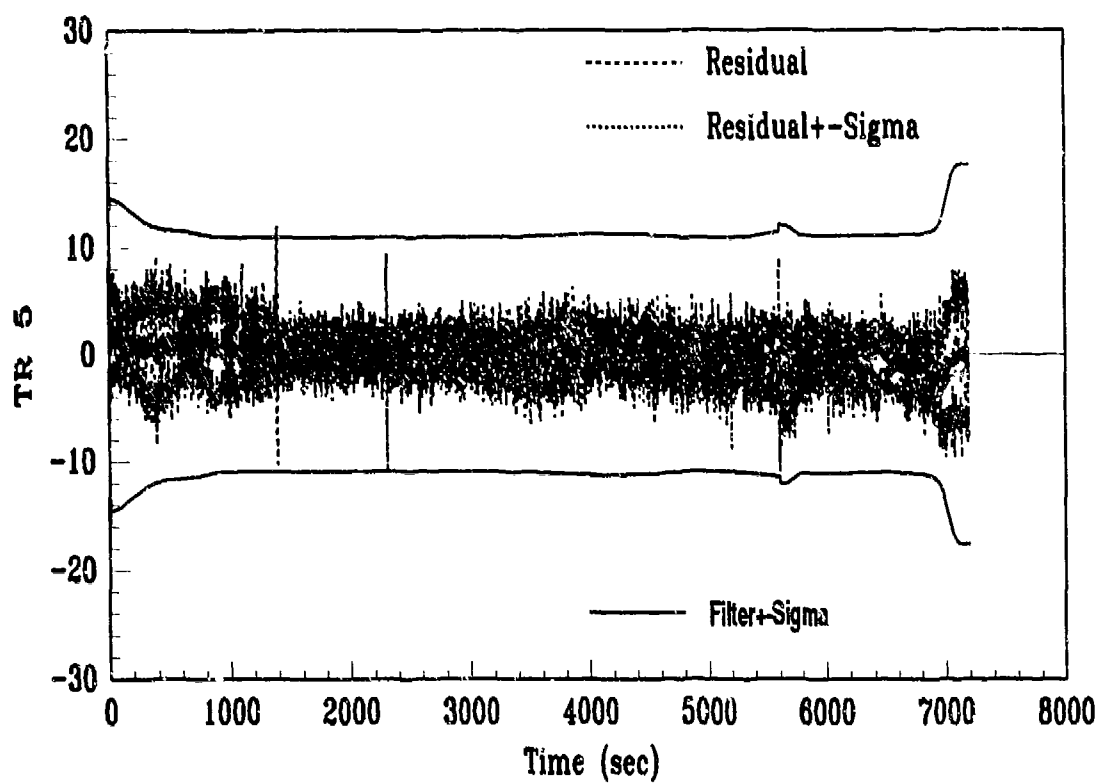


Figure D.20. Residual Plots: Baseline

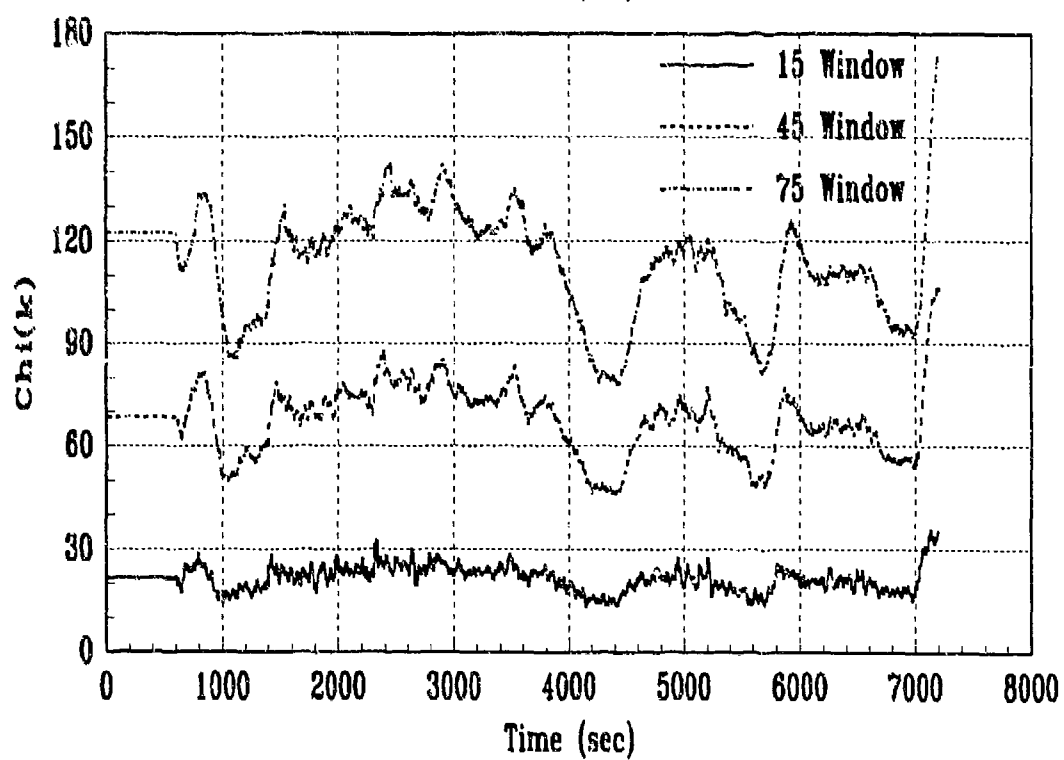
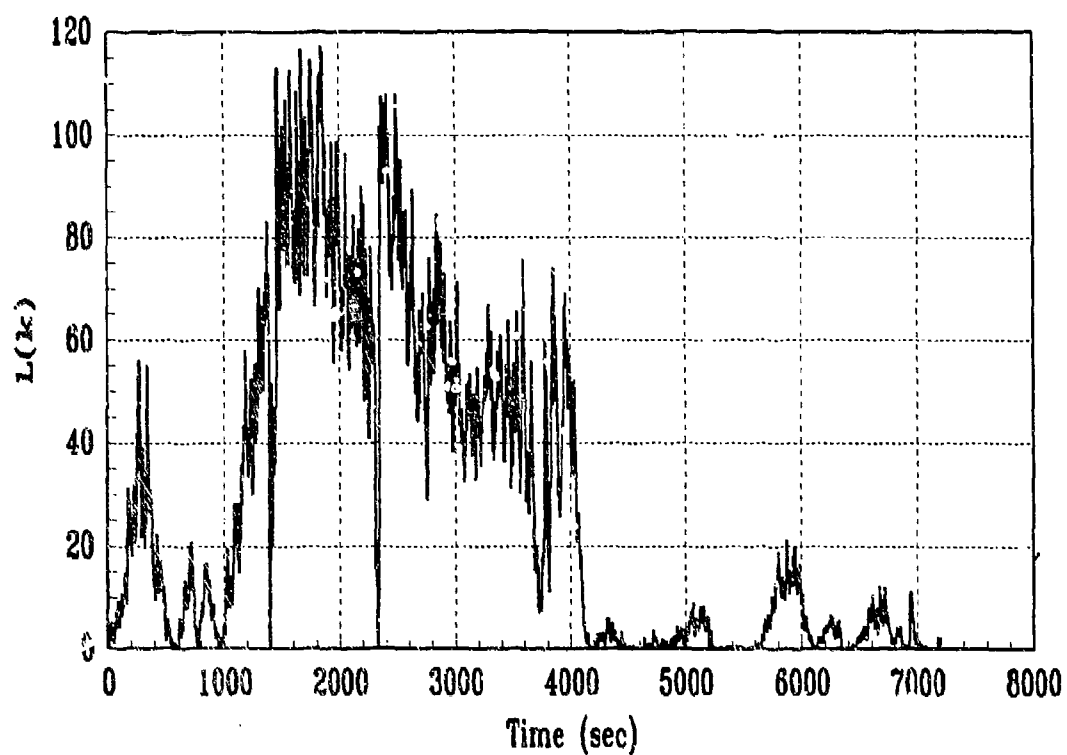


Figure D.21. GLR/CHI: Baseline

Appendix E. *Jamming Failure Plots*

Selected state and residual plots are shown to illustrate the performance of the NRS filter under heavy jamming conditions. GLR and CHI plots are presented for all three jamming levels.

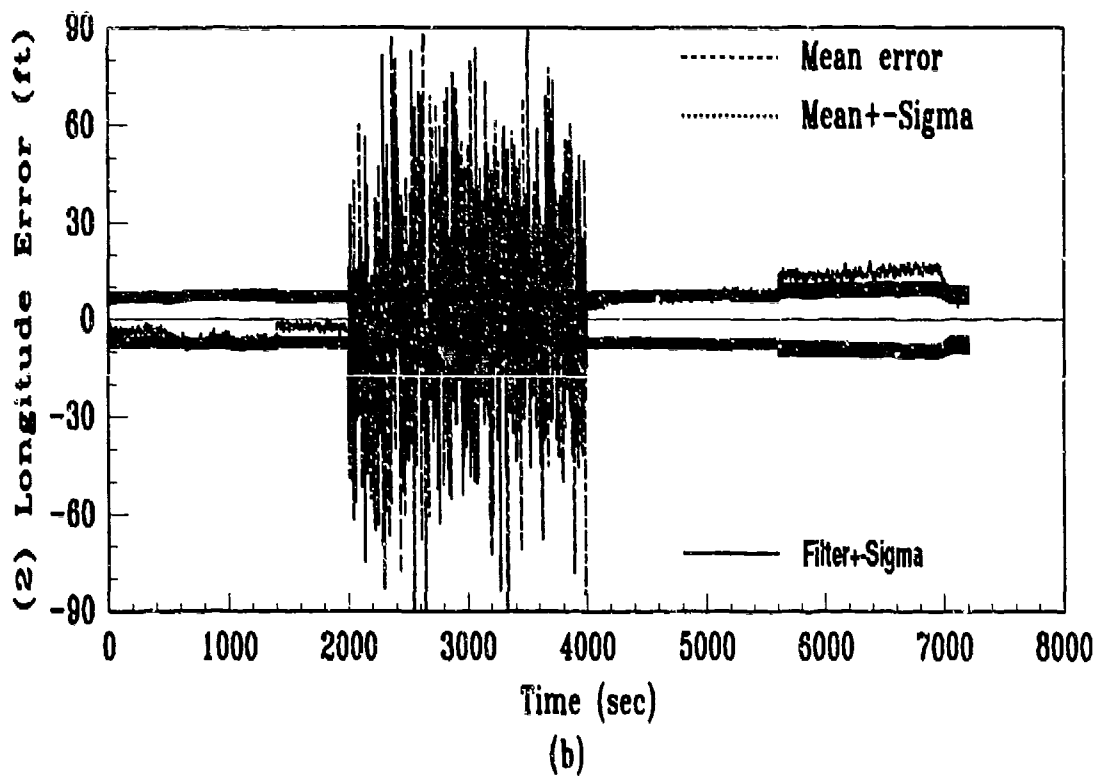
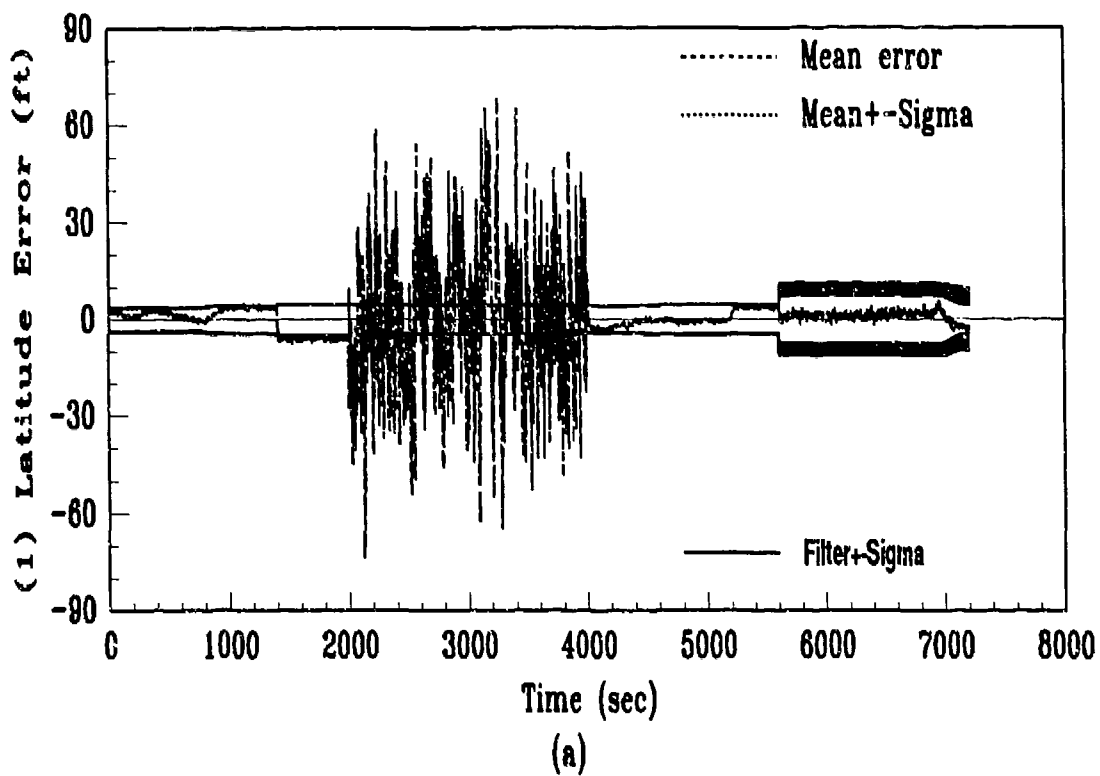


Figure E.1. State Plots: Heavy Jamming

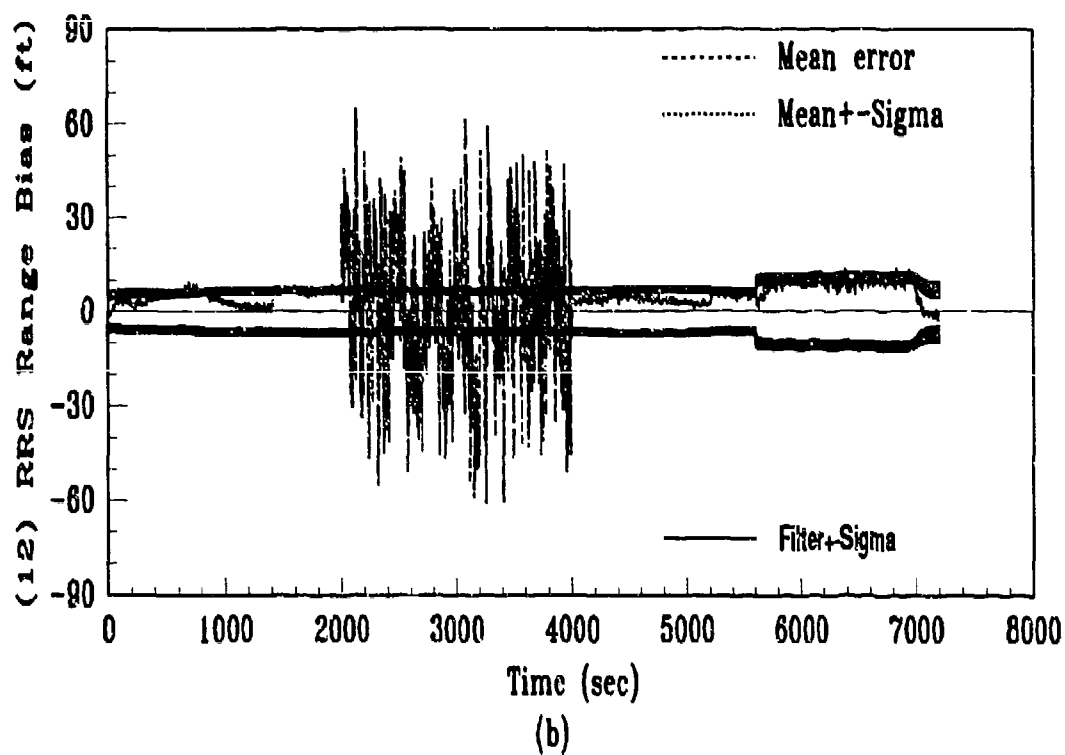
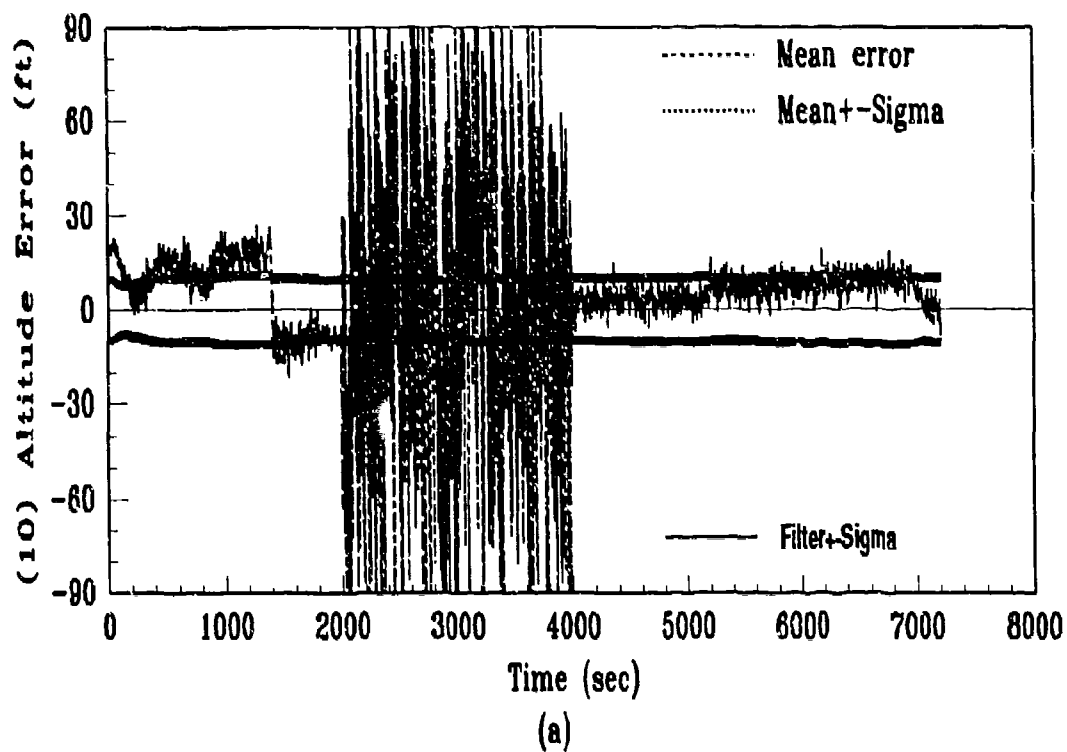


Figure E.2. State Plots: Heavy Jamming

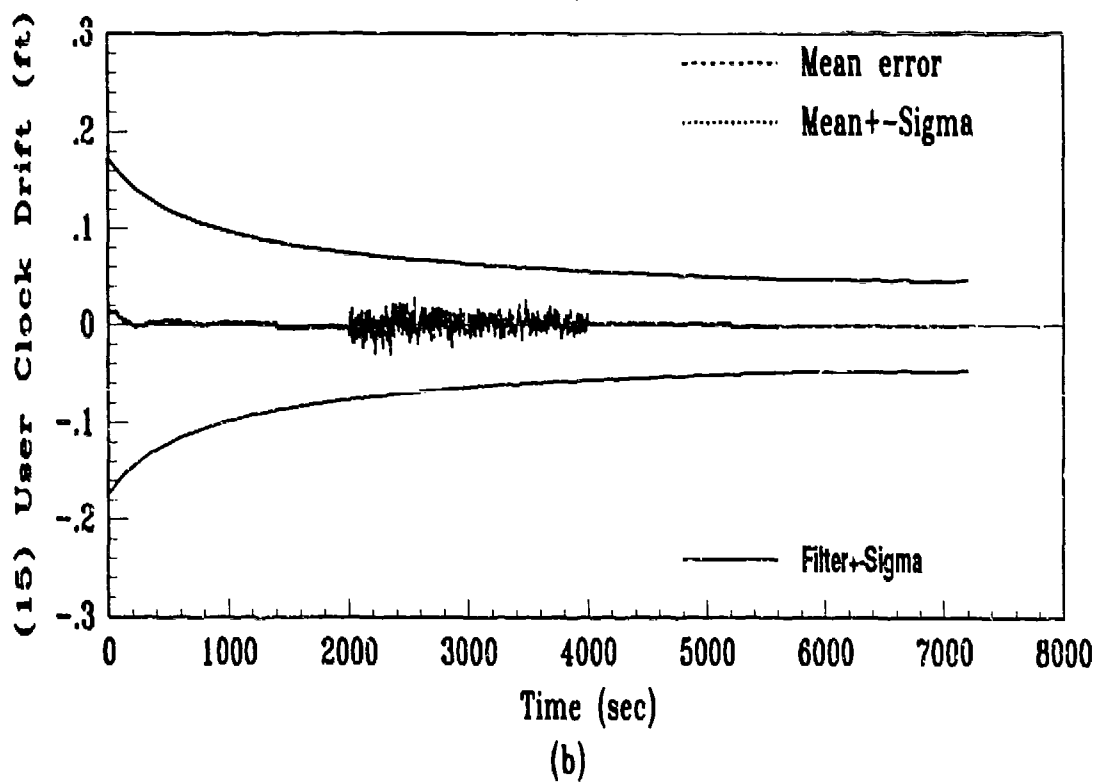
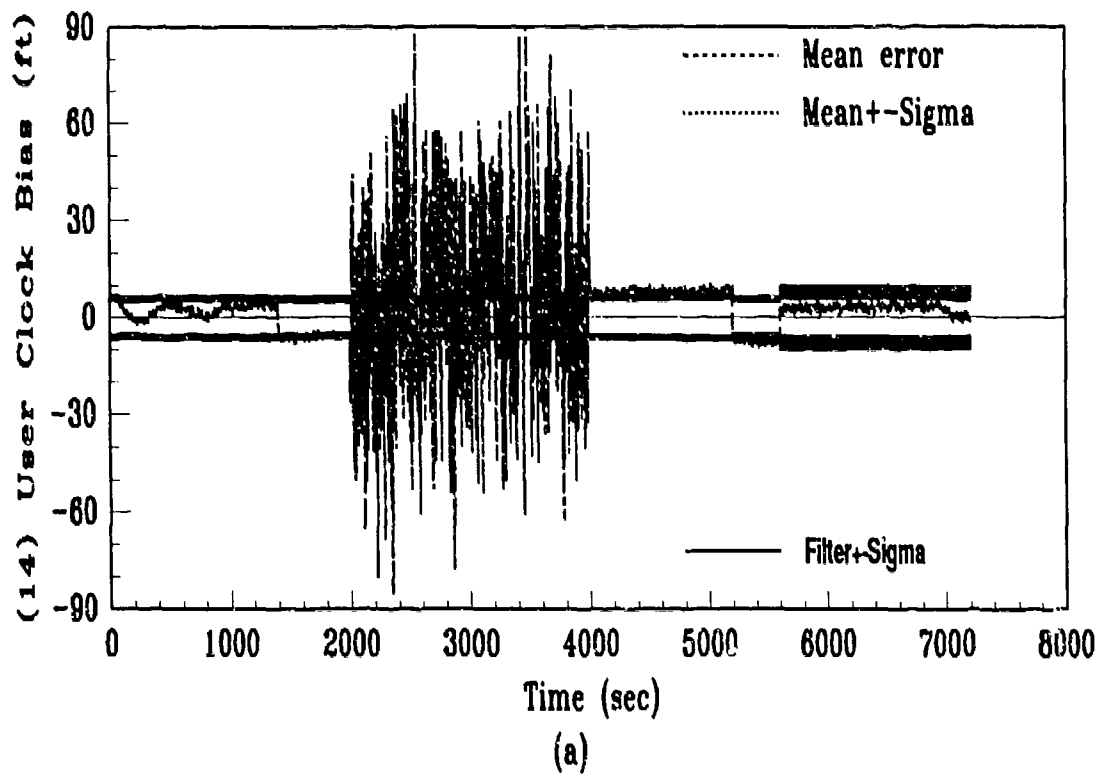
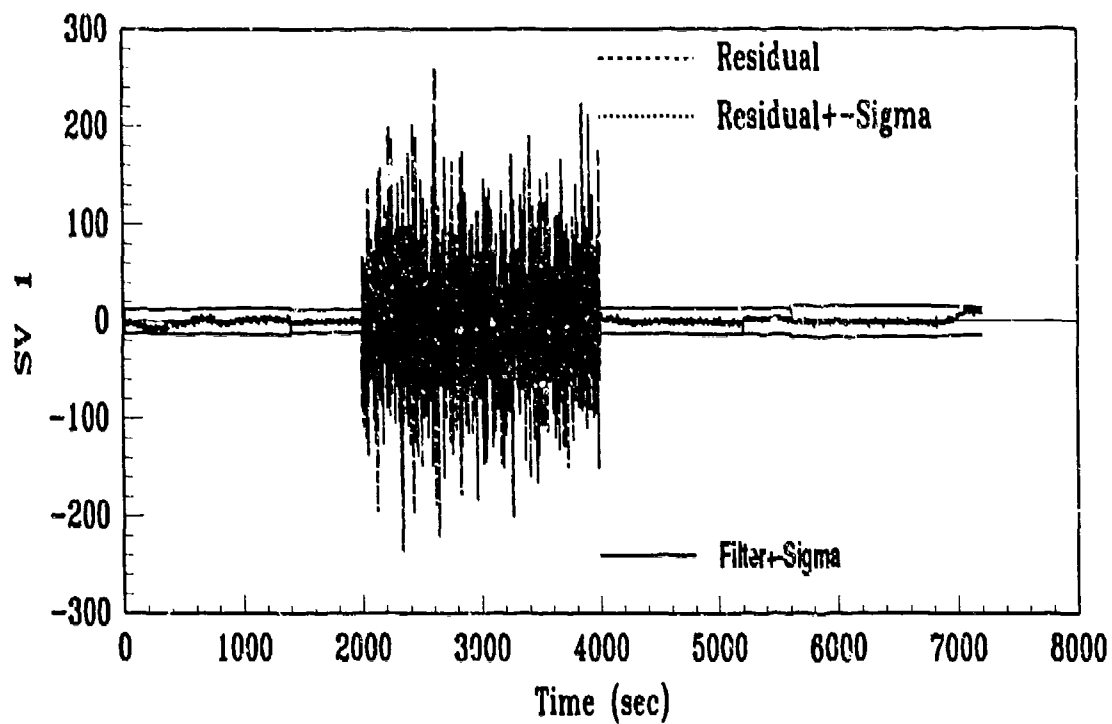
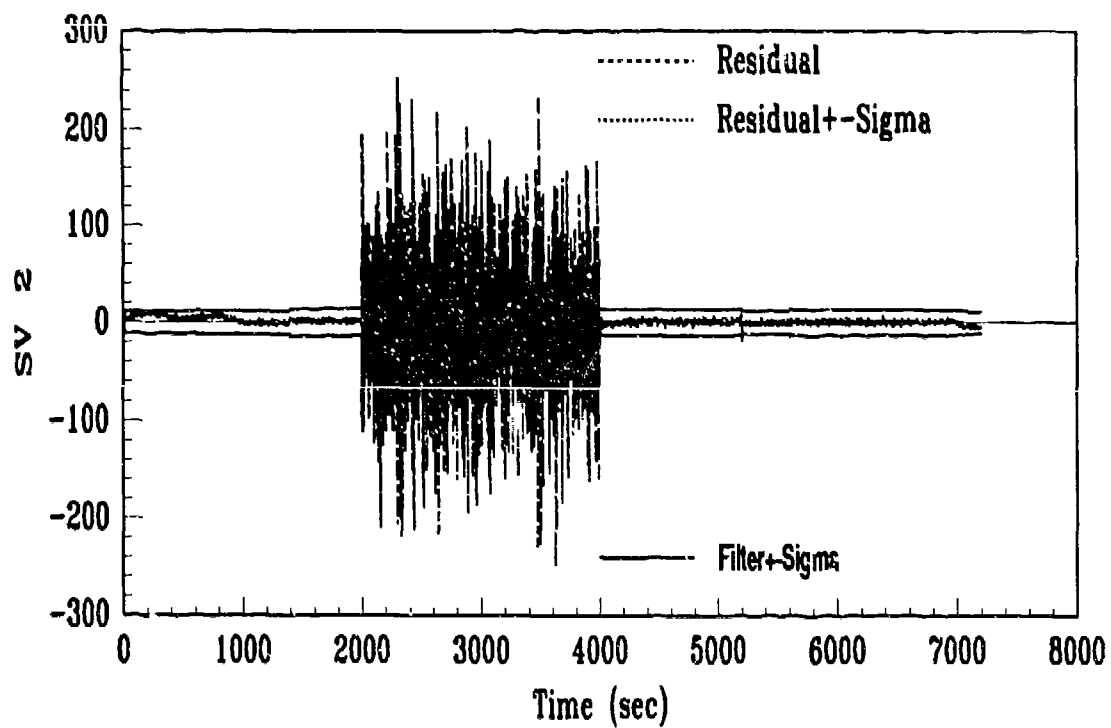


Figure E.3. State Plots: Heavy Jamming



(a)



(b)

Figure E.4. Residual Plots: Heavy Jamming

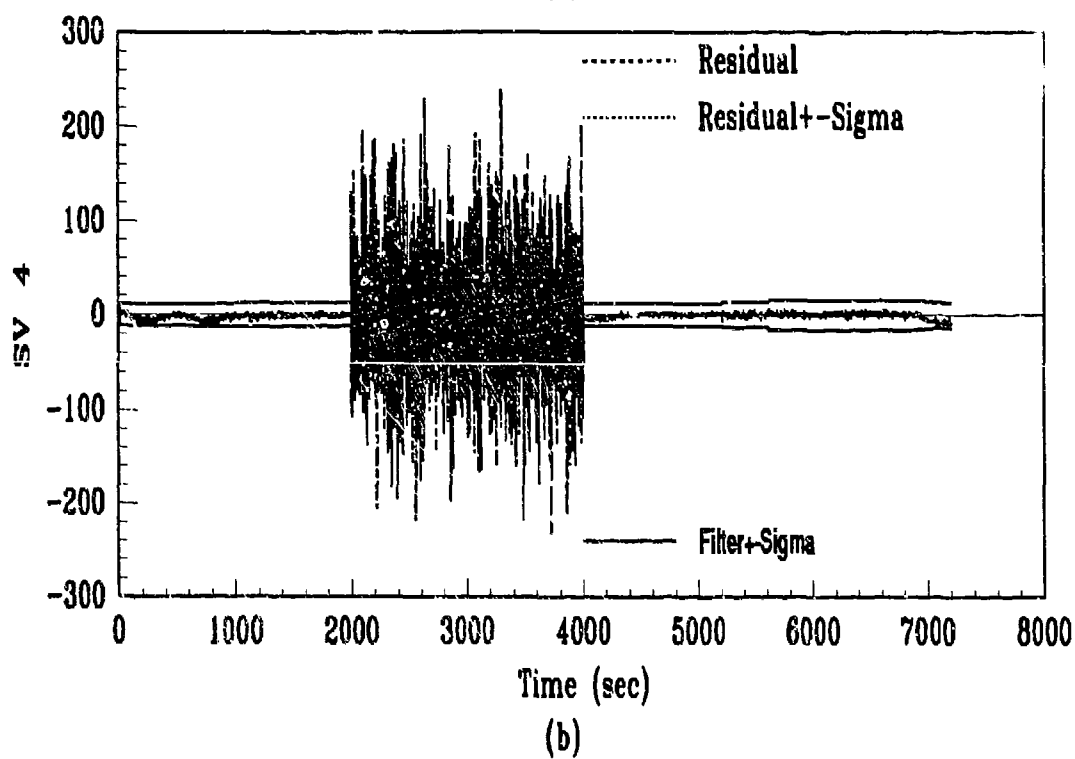
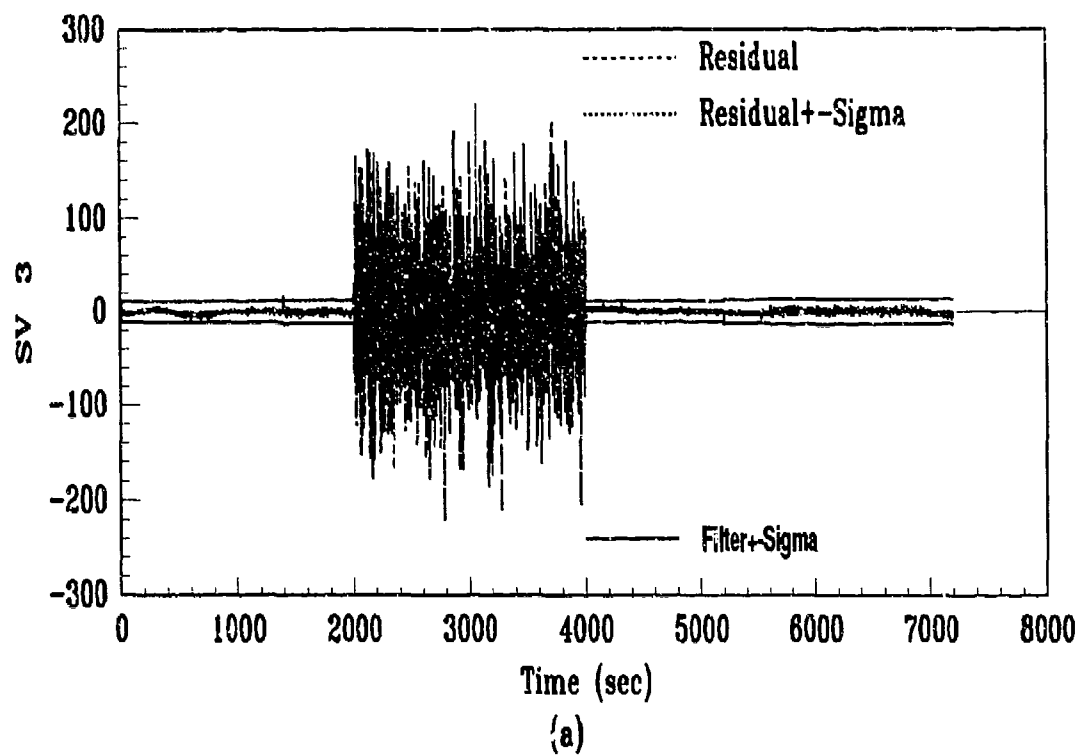


Figure E.5. Residual Plots: Heavy Jamming

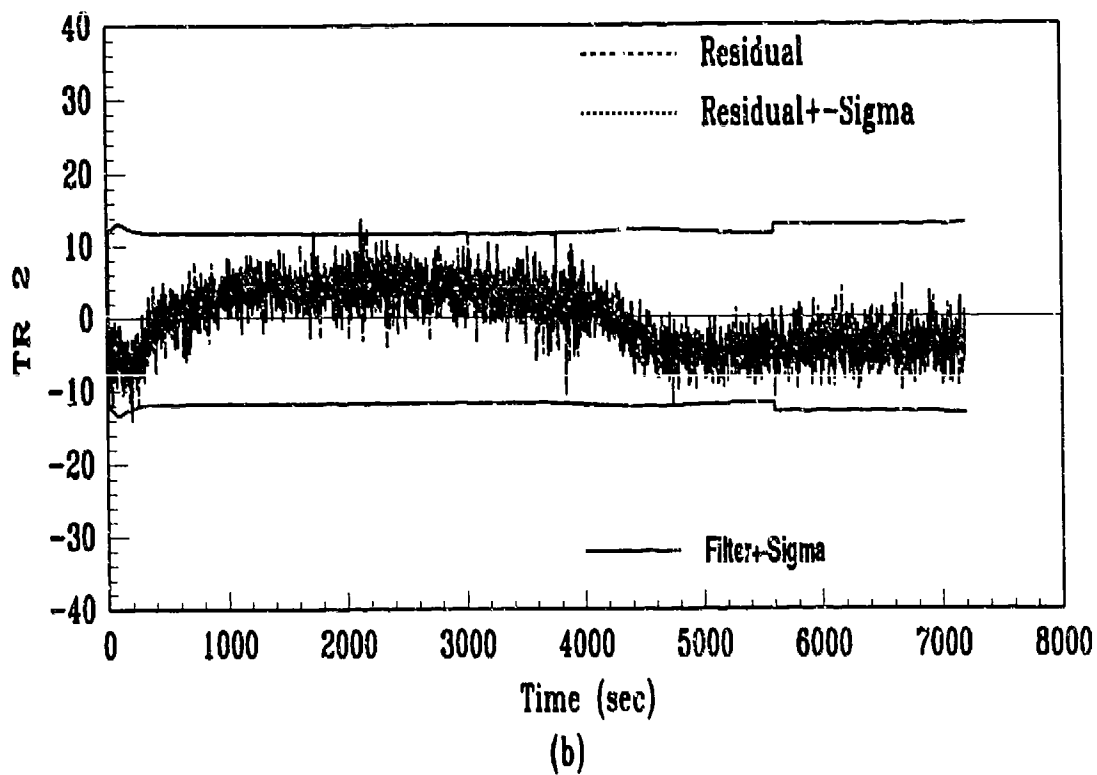
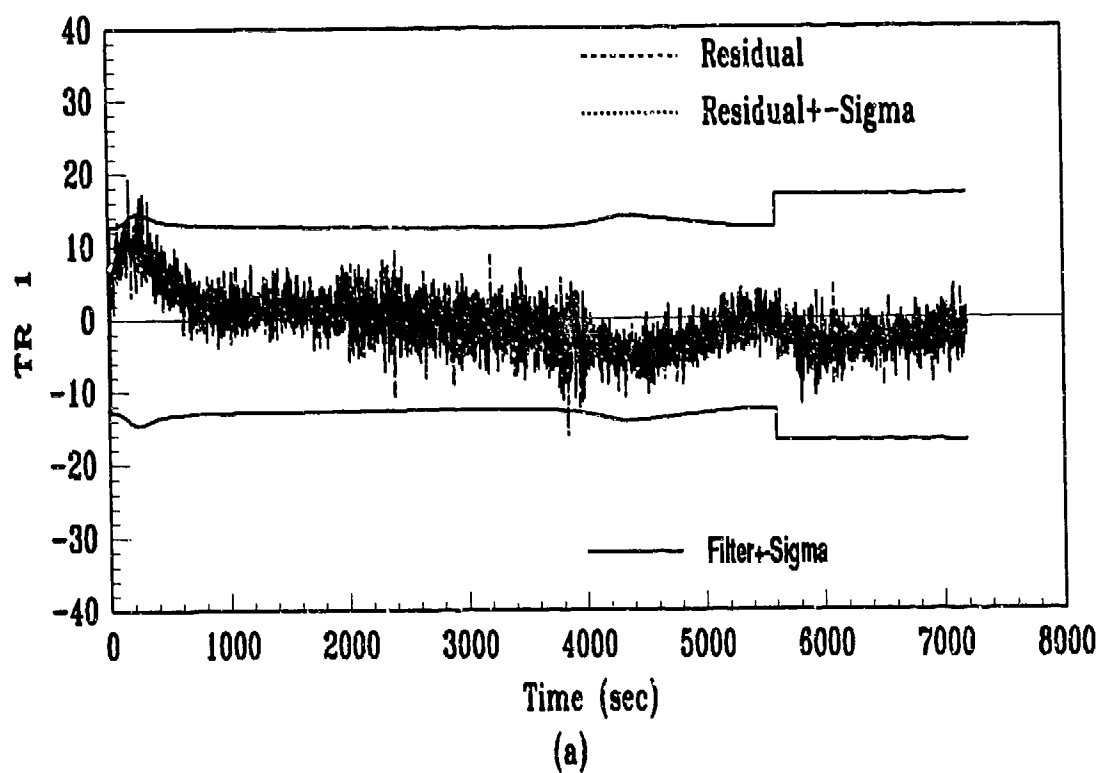


Figure E.6. Residual Plots: Heavy Jamming

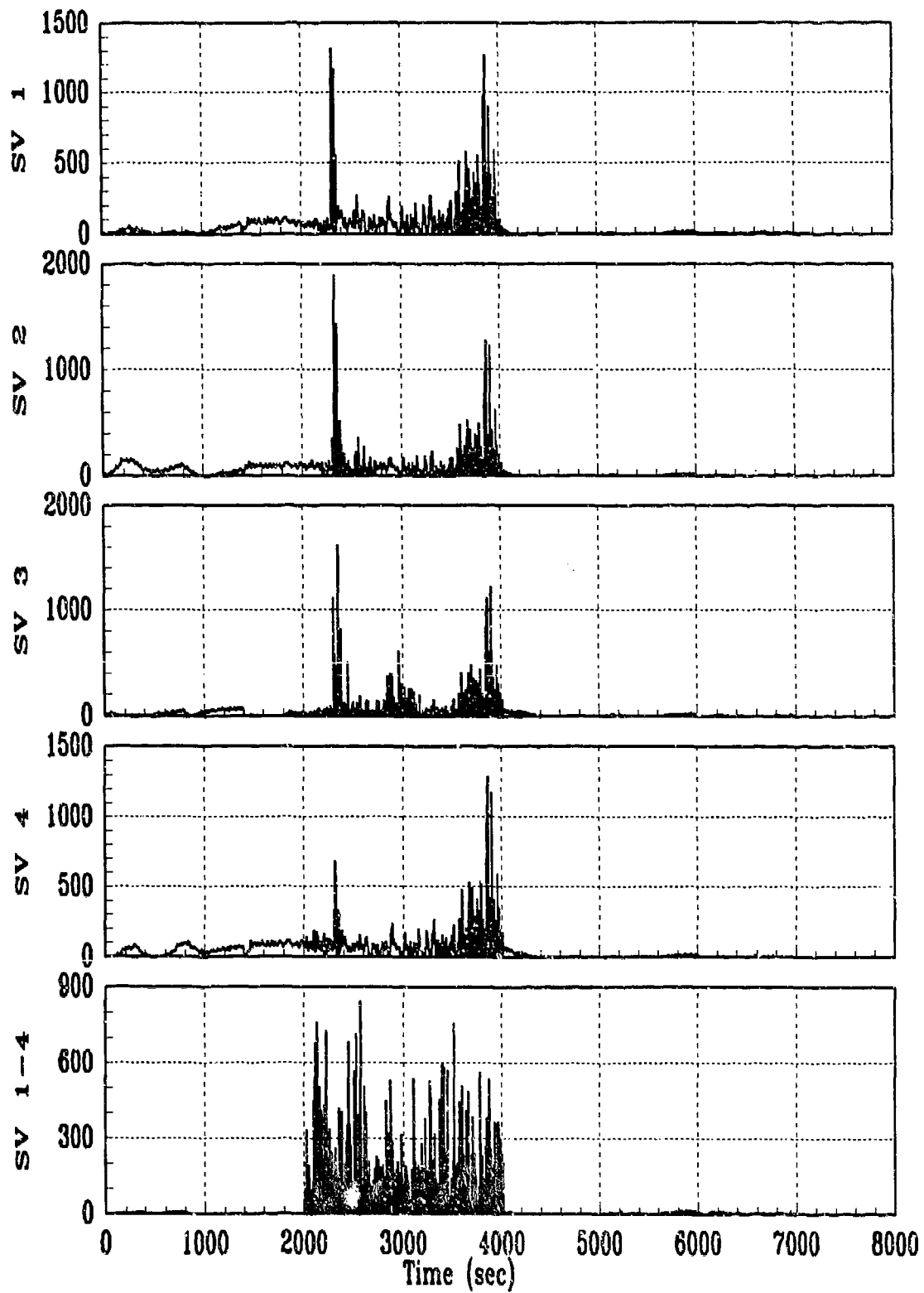
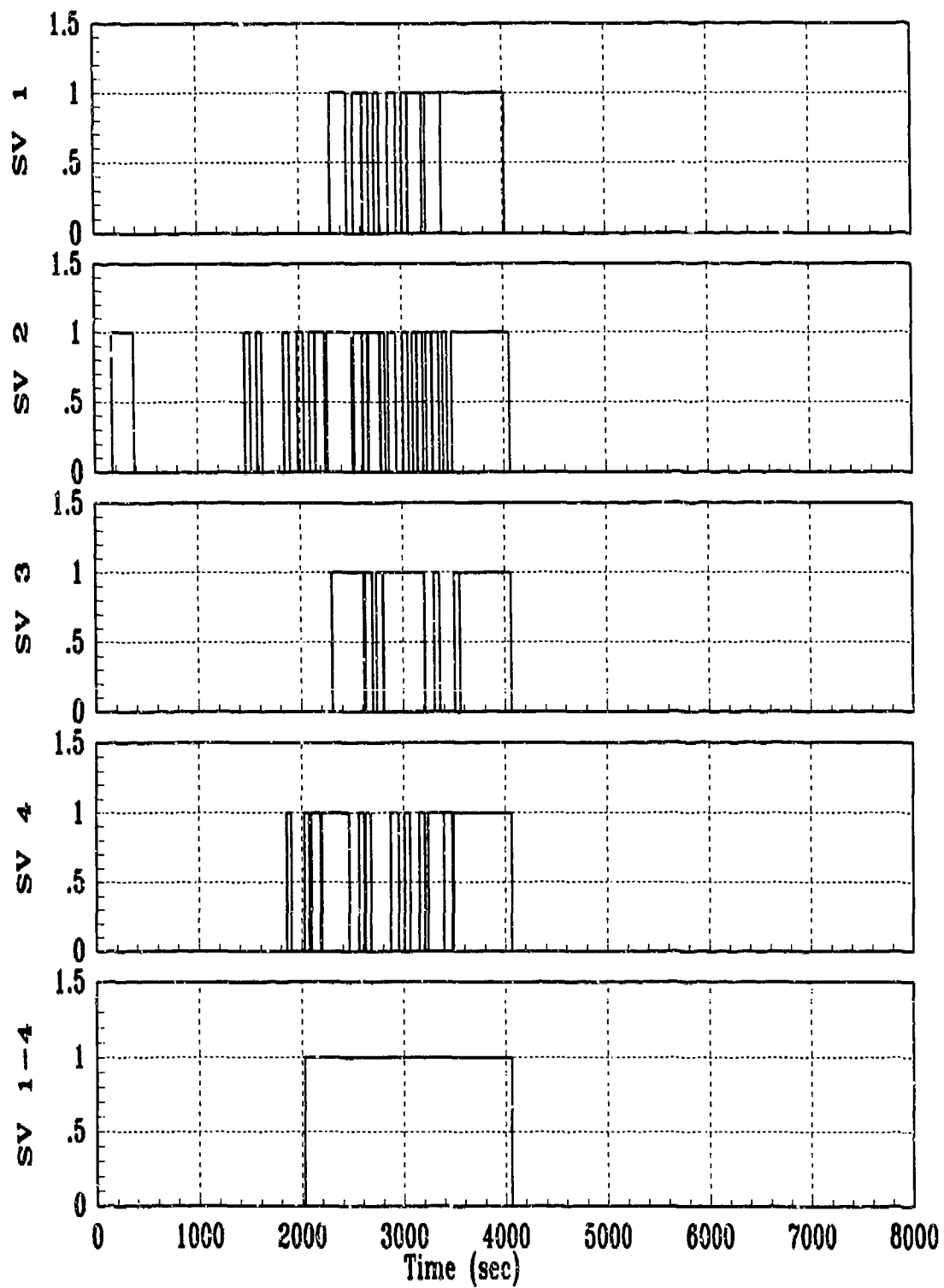


Figure E.7. GLR: Heavy Jamming



Threshold=120 , Number Low=23

Figure E.8. Fail Flag: Heavy Jamming

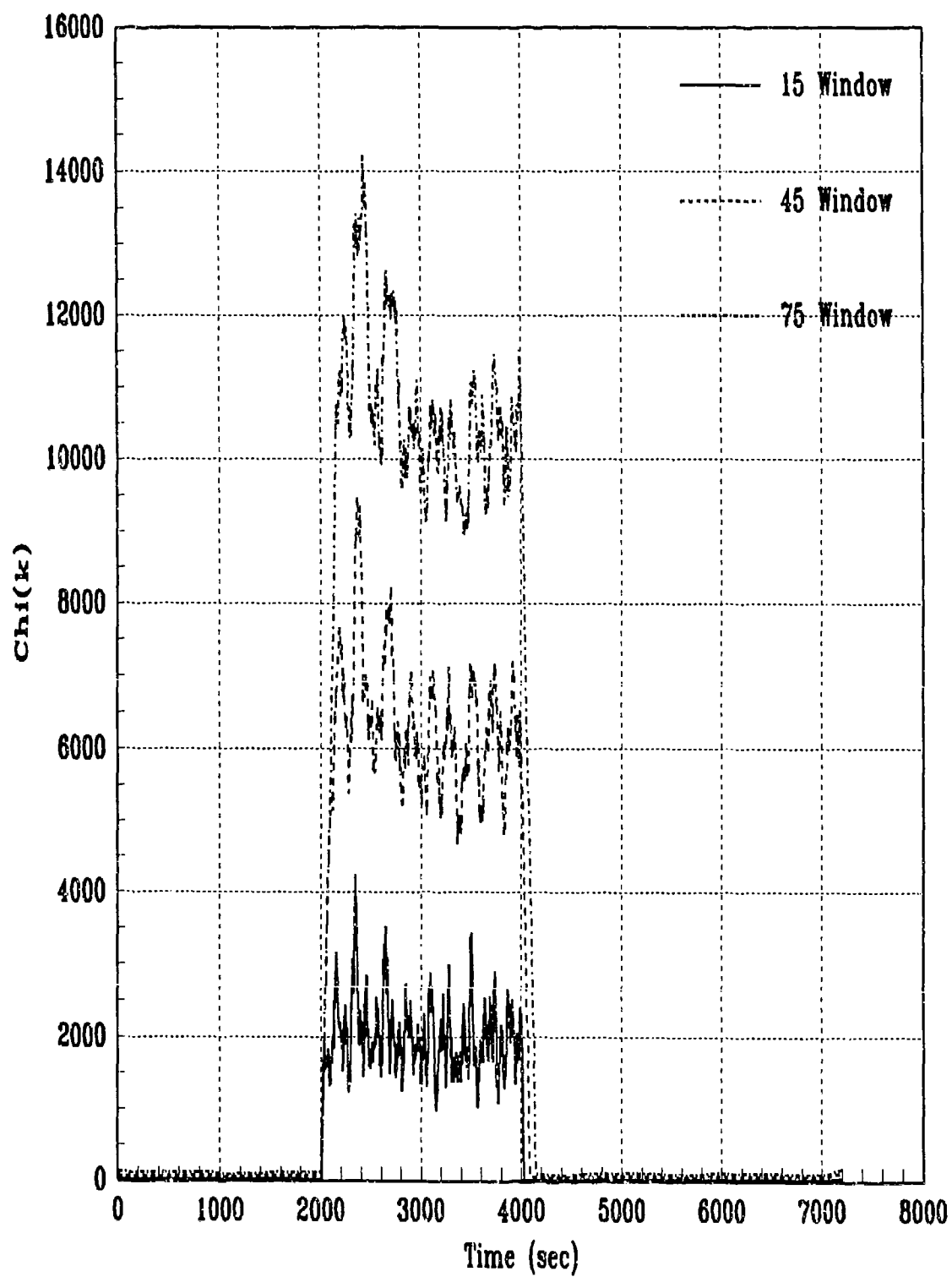


Figure E.9. CHI: Heavy Jamming

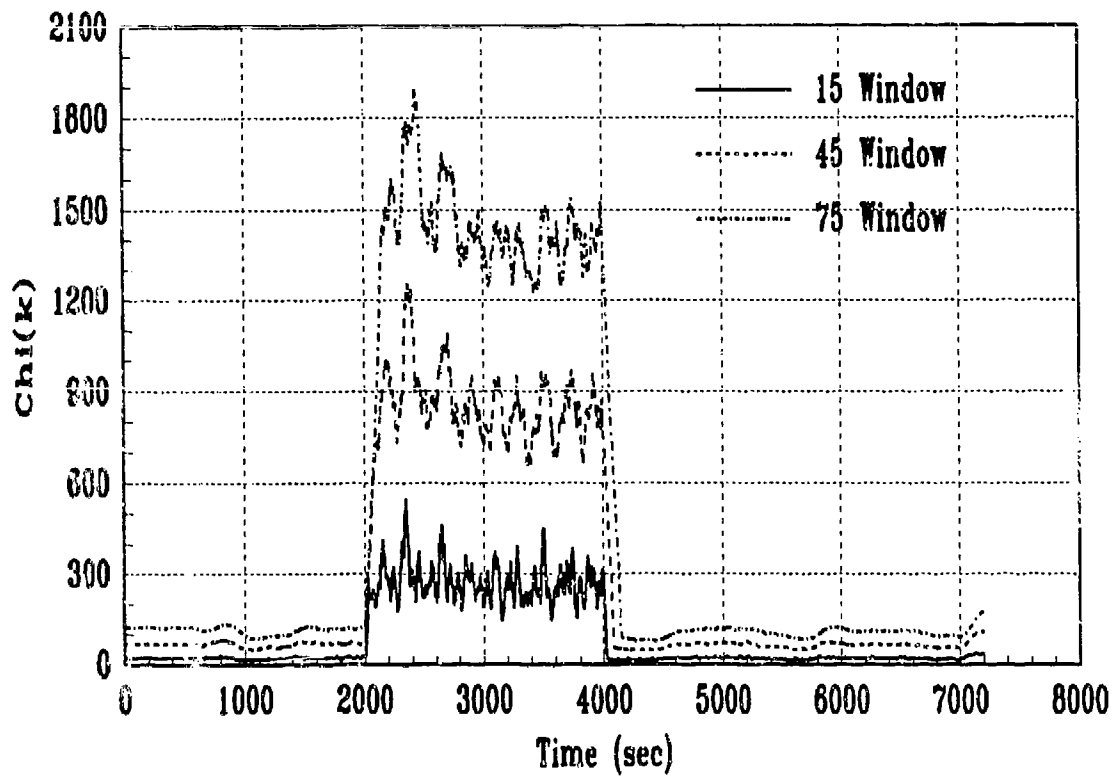
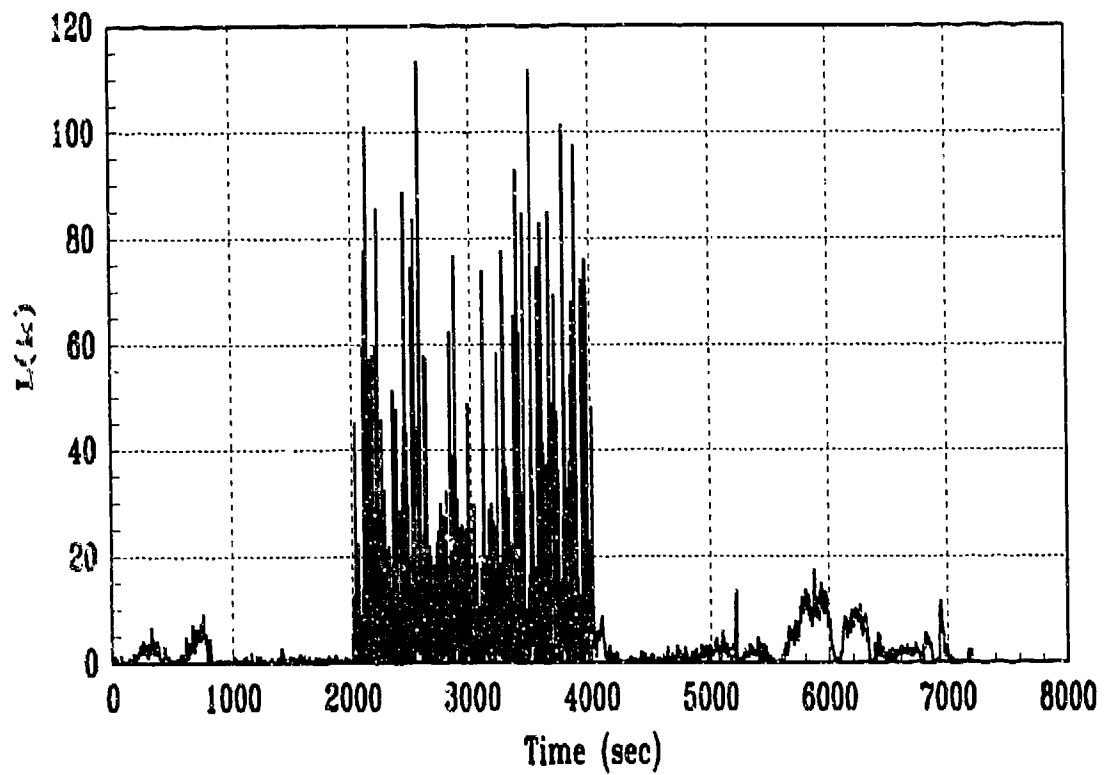


Figure E.10. GLR/ χ : Medium Jamming

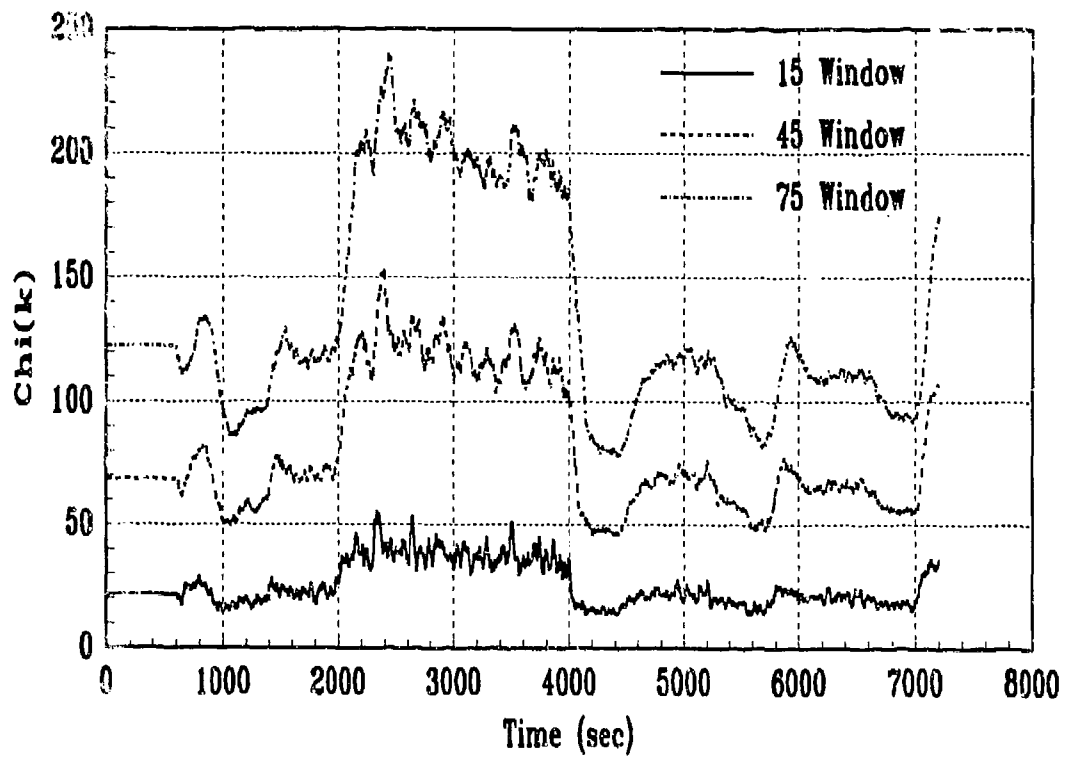
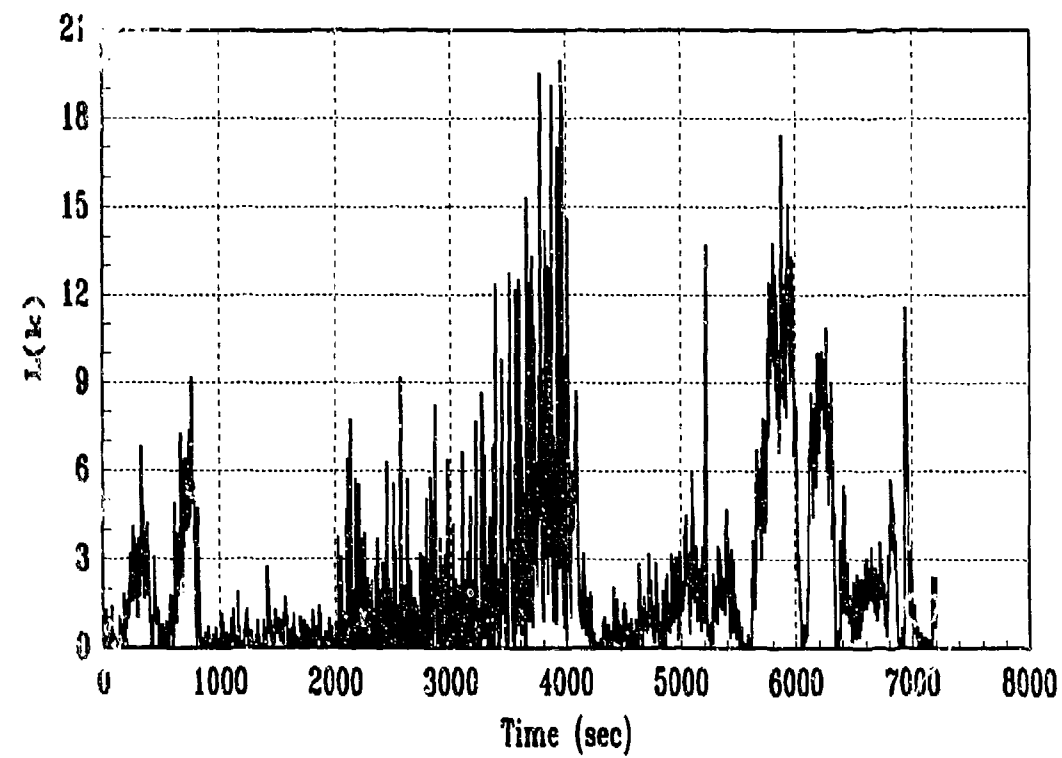


Figure E.11. GLR/CHI: Light Jamming

Appendix F. *Bias Failure Plots*

Selected state and residual plots are shown to illustrate the performance of the NRS filter with bias failures induced on the SV's. Some of the statistics are masked by the large vertical scale of the plots, but this allows the full range of the mean error and residual values to be seen, which best characterizes the filter performance. A variety of GLR and χ plots illustrate the FDI performance for all bias failures considered. Several plots are shown twice with two different vertical scales to reveal significant details that are lost in other plots.

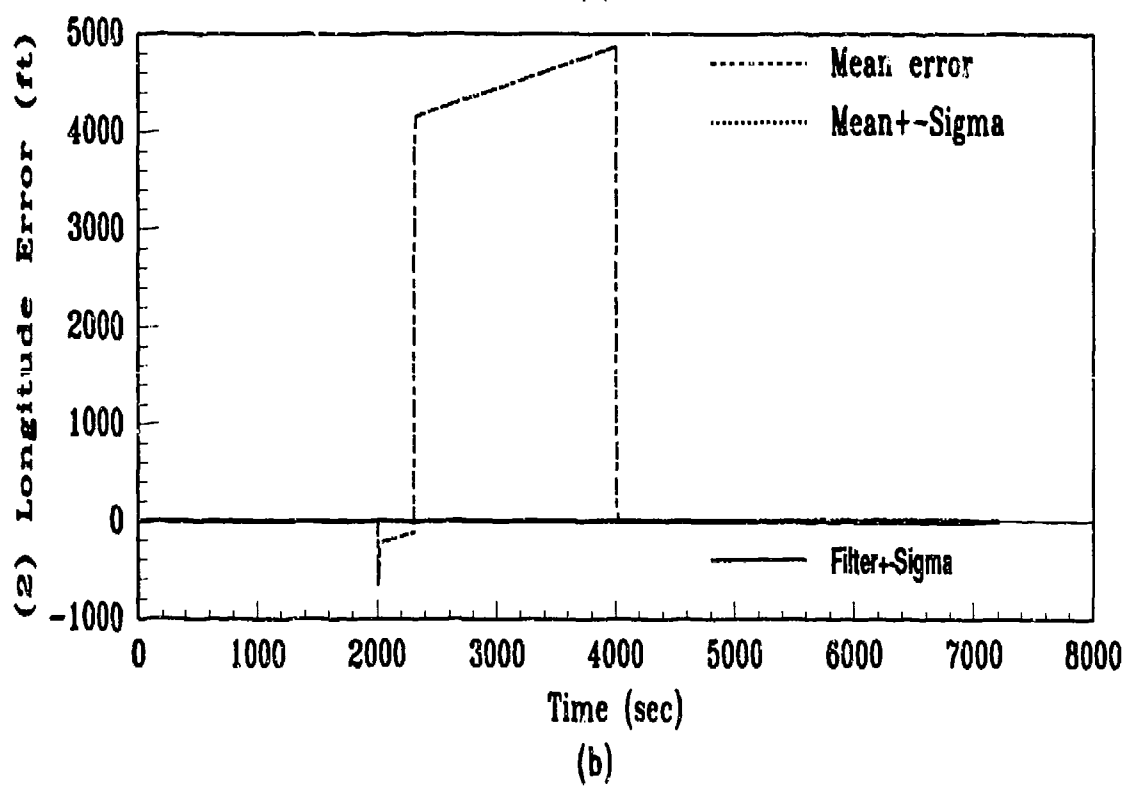
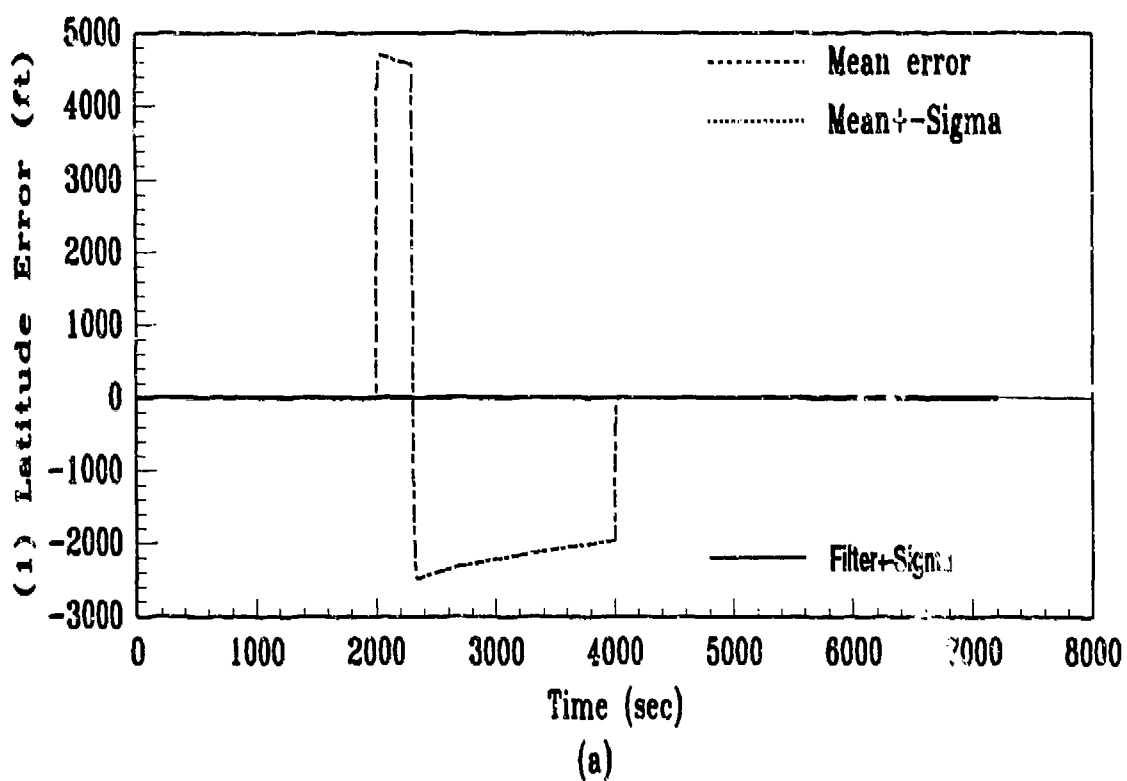


Figure F.1. State Plots: Run 4, Bias=7000

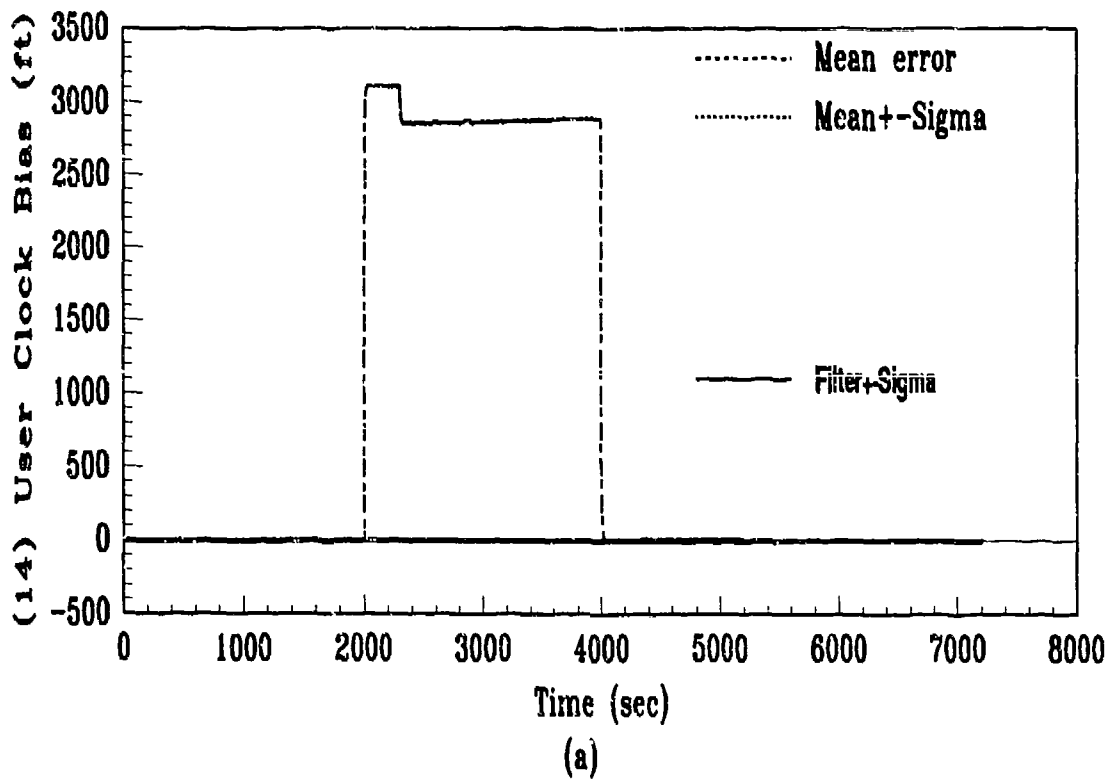
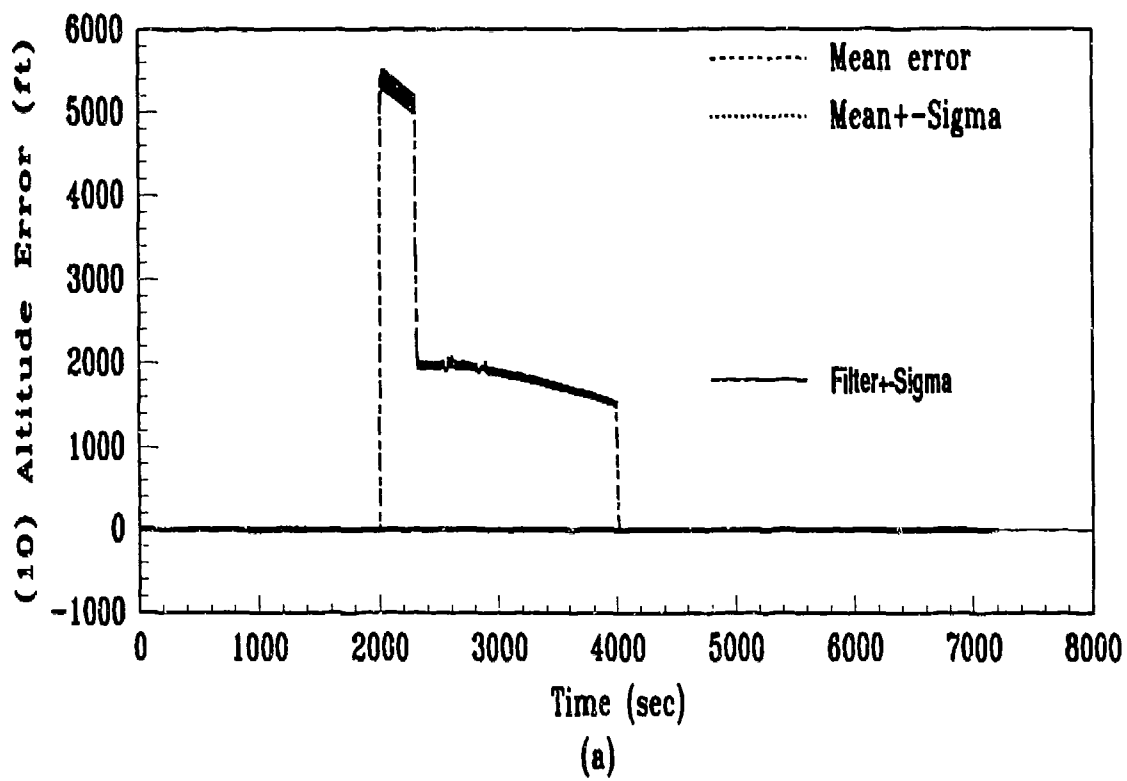
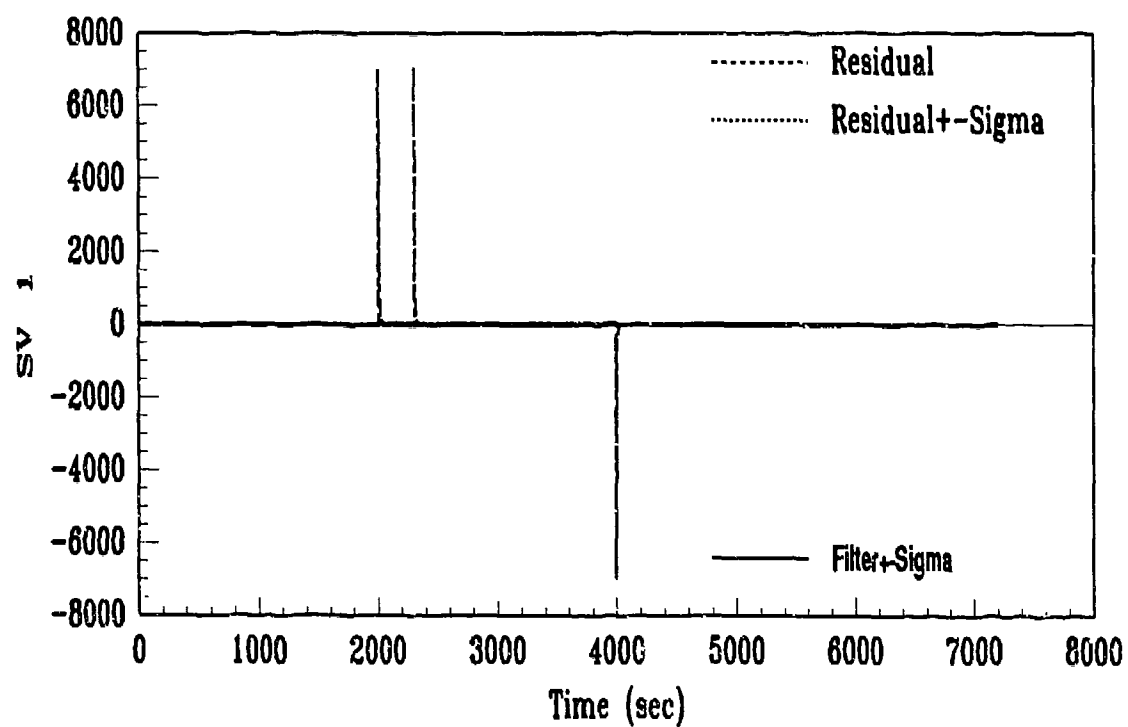
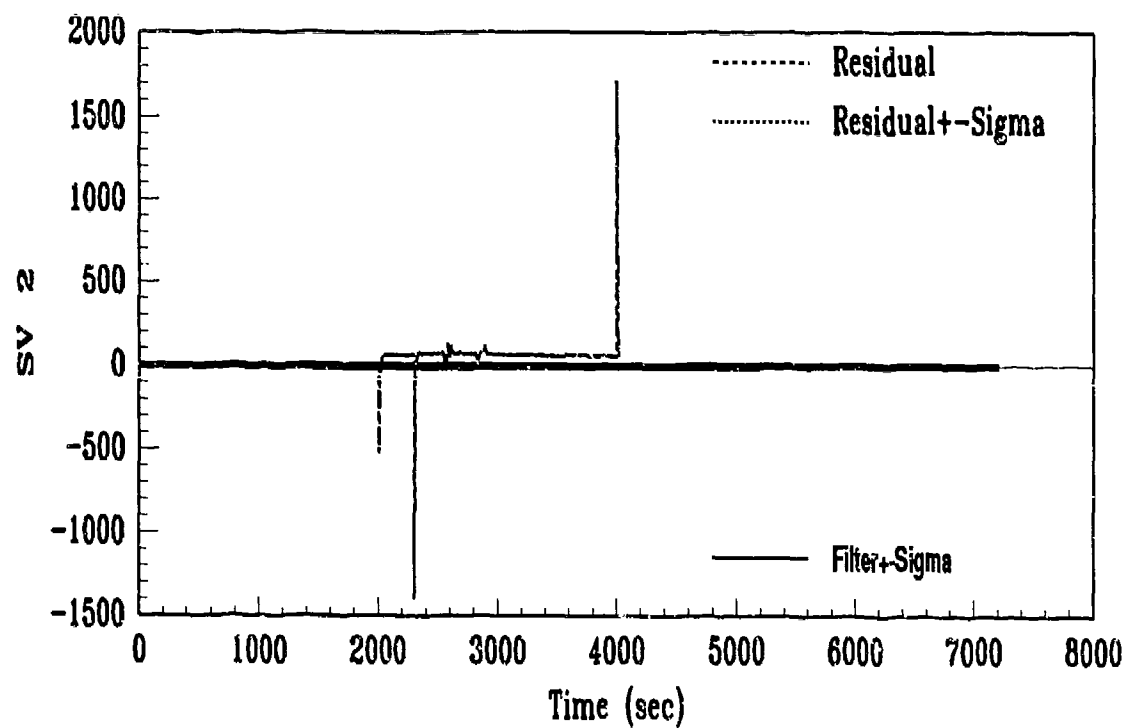


Figure F.2. State Plots: Run 4, Bias=7000



(a)



(b)

Figure F.3. Residual Plots: Run 4, Bias=7000

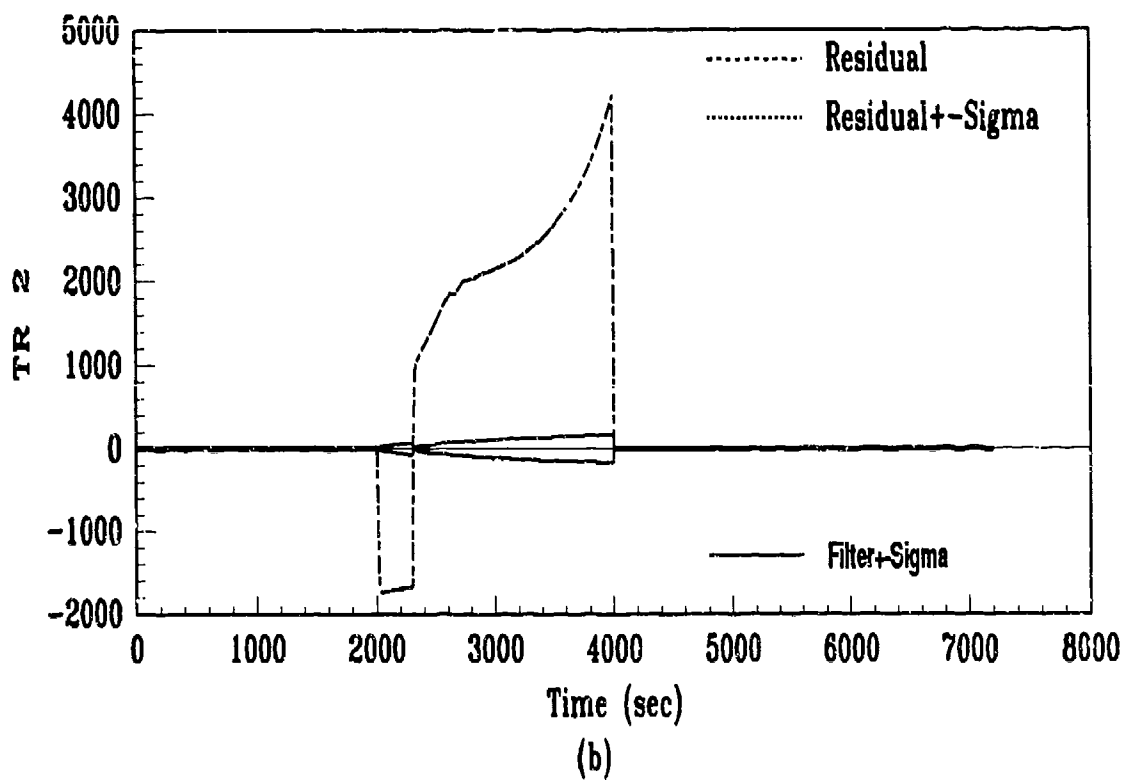
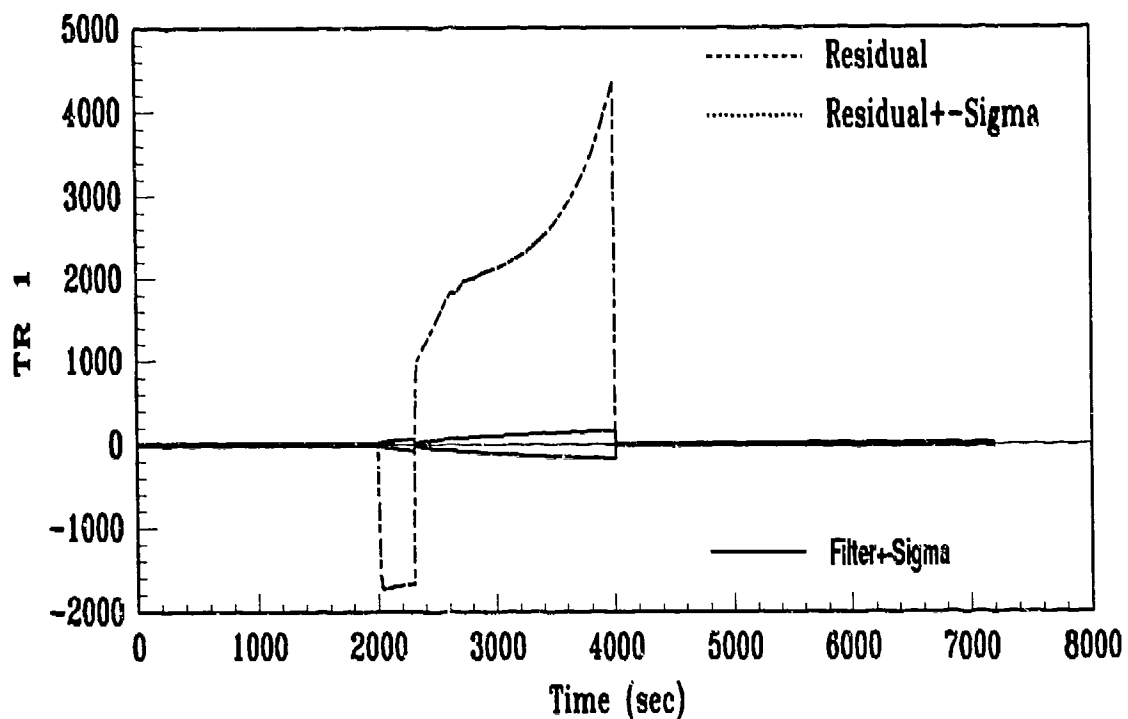


Figure F.4. Residual Plots: Run 4, Bias=7000

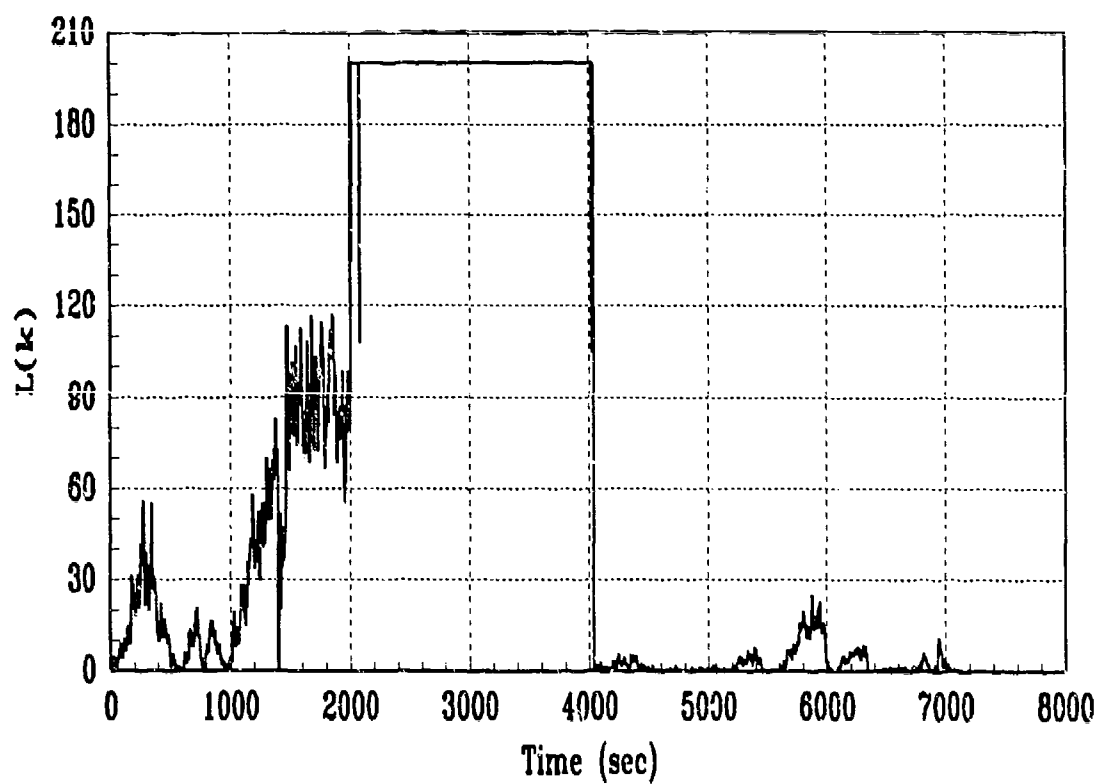
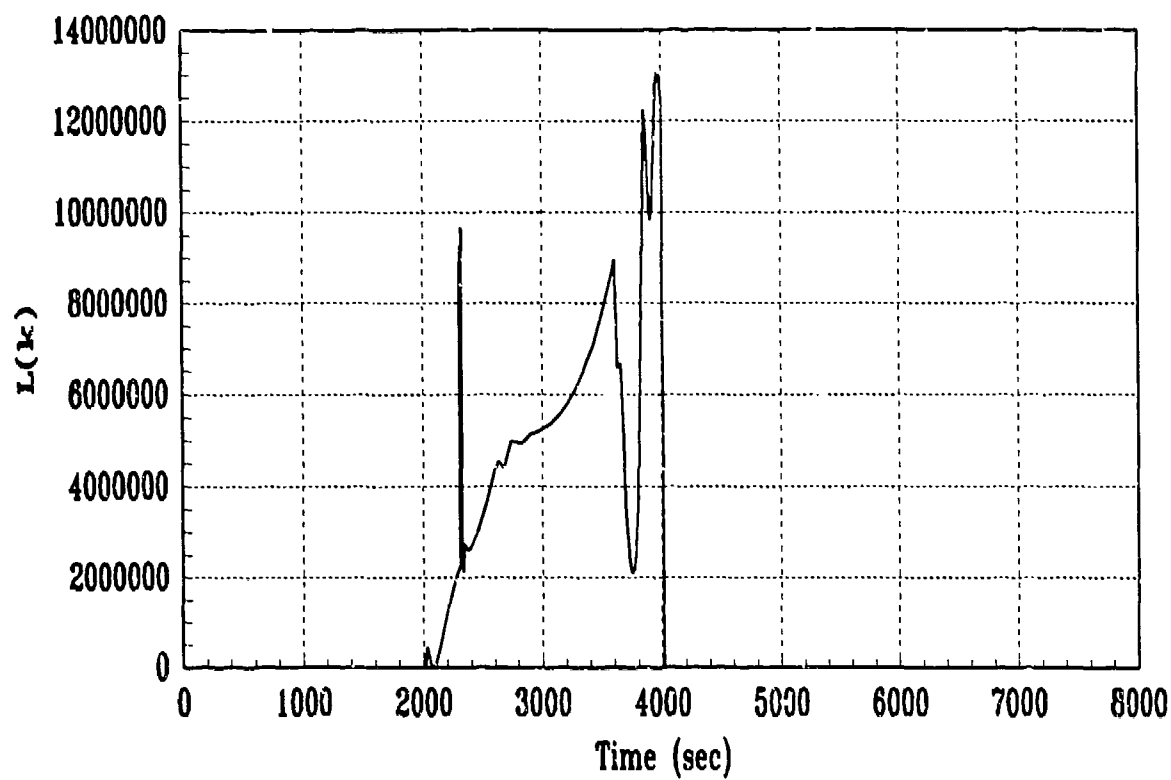


Figure F.5. GLR: Run 4, Bias=7000

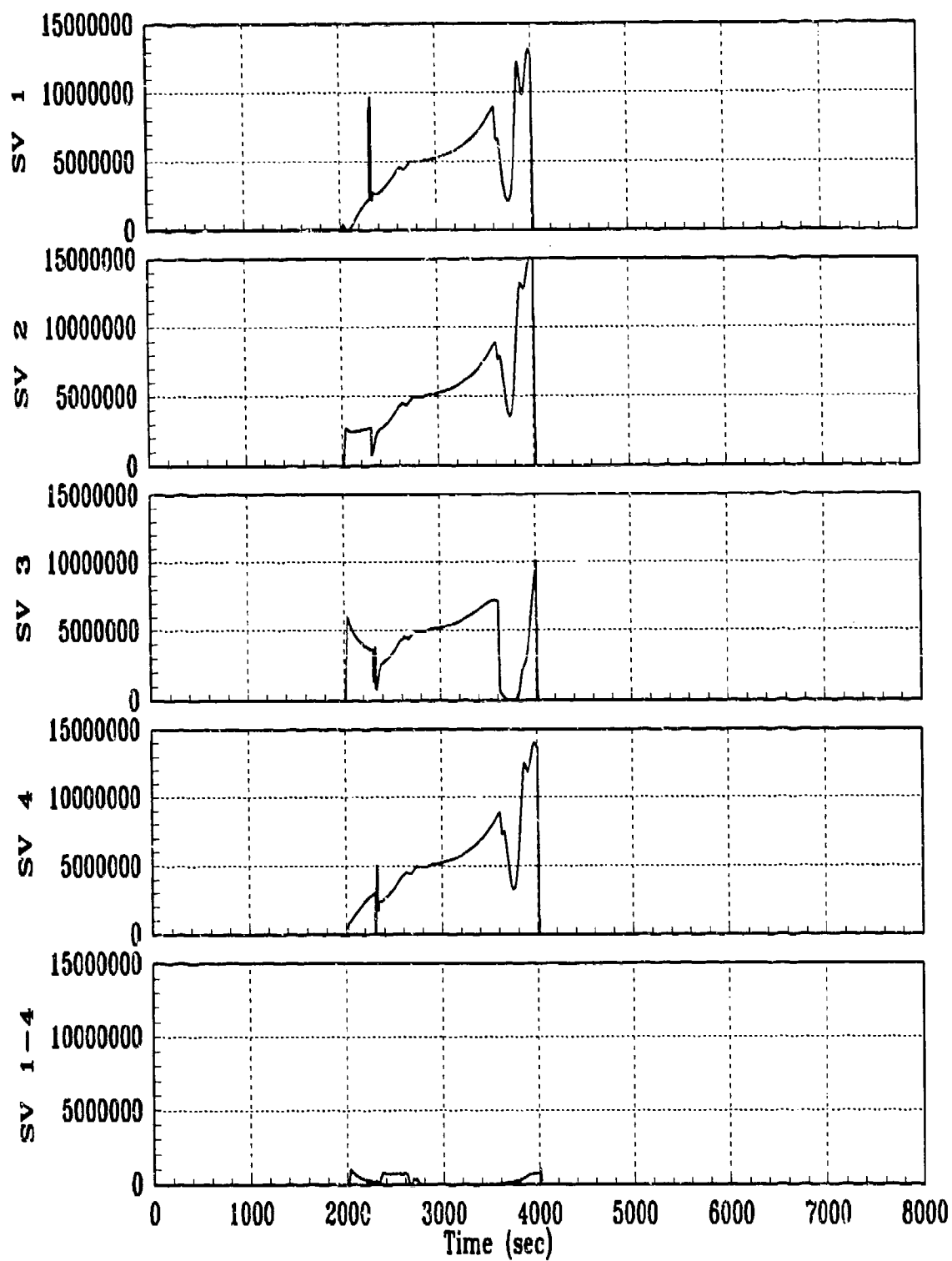
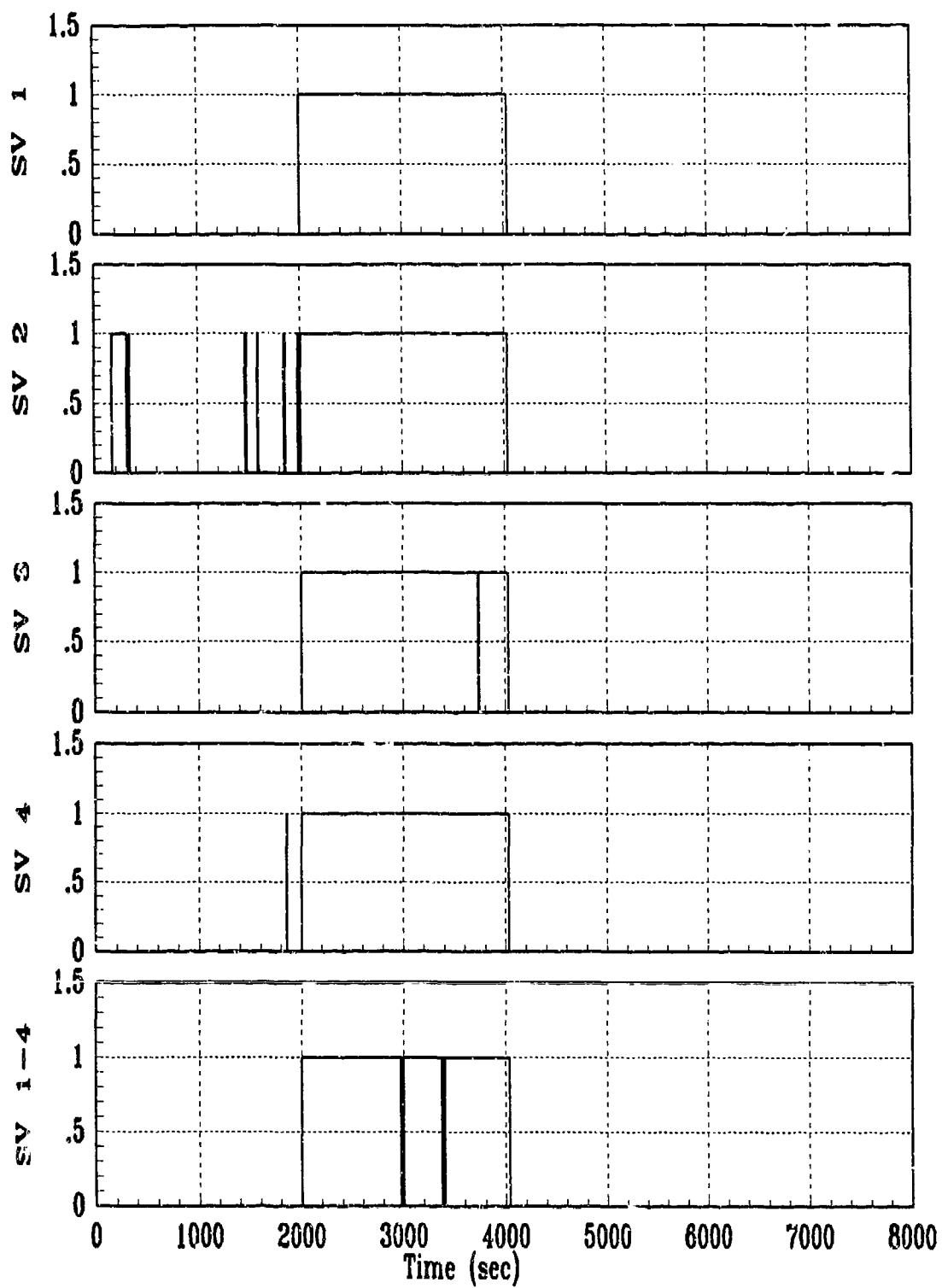


Figure F.6. GLR: Run 4, Bias=7000



Threshold=120 , Number Low=2

Figure F.7. Fail Flag: Run 4, Bias=7000

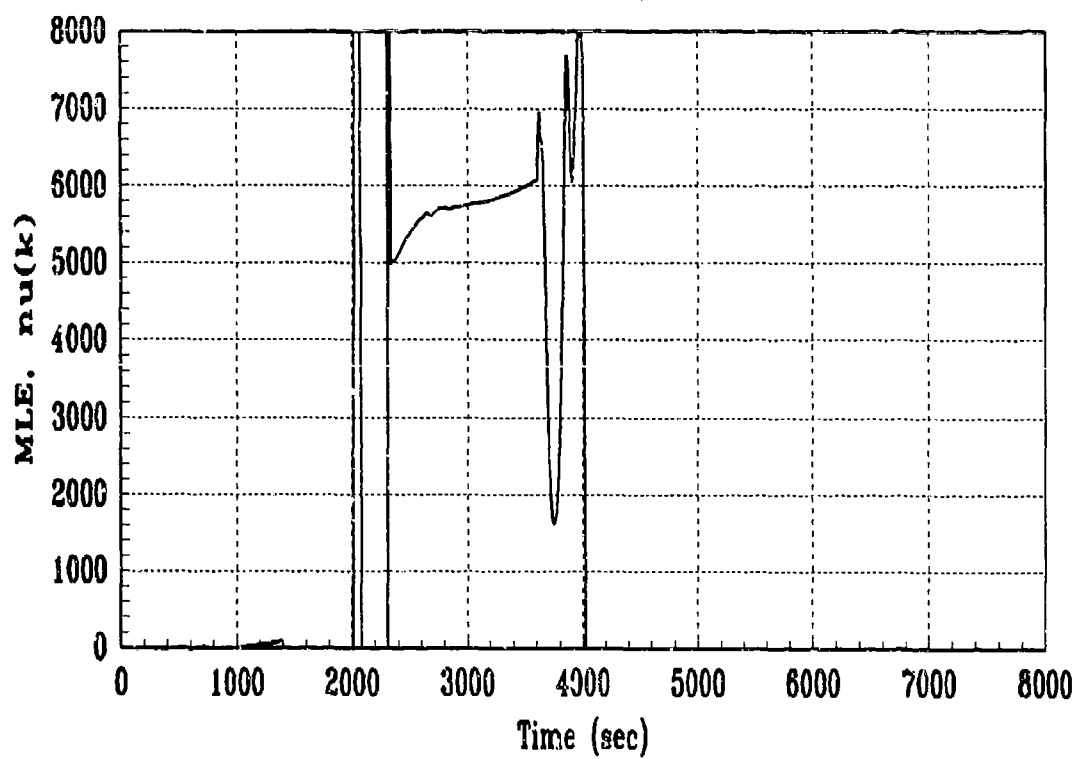
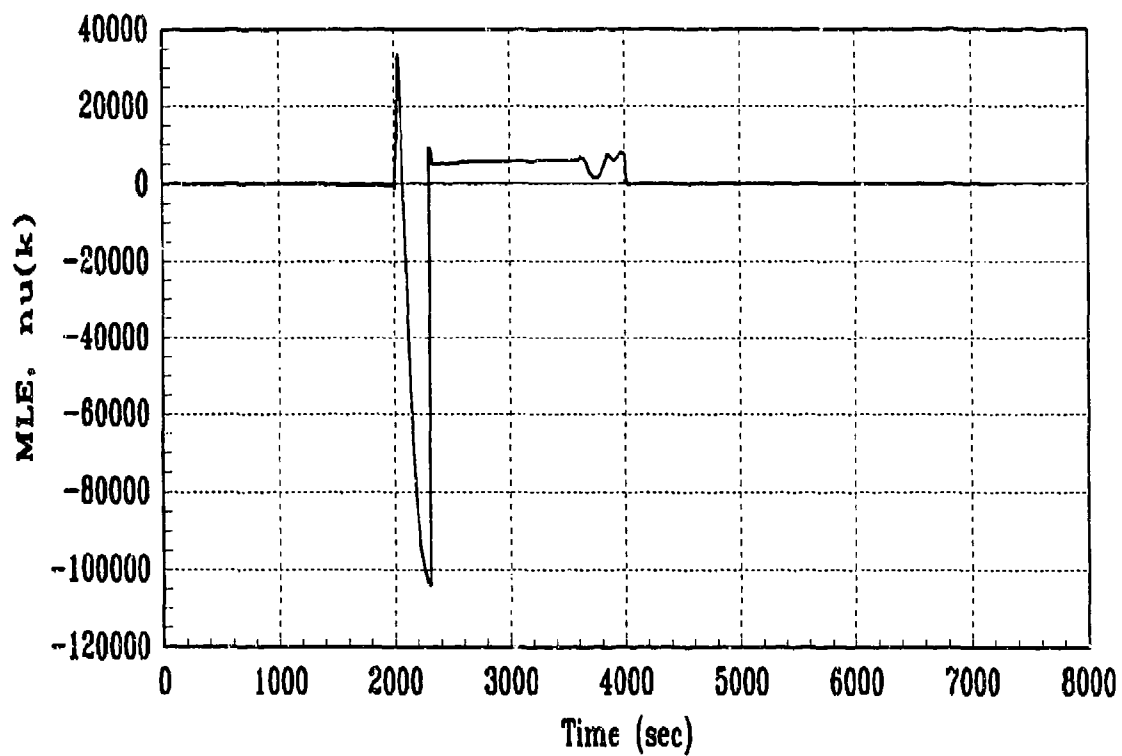


Figure F.8. MLE $\hat{\nu}(t_k)$: Run 4, Bias=7000

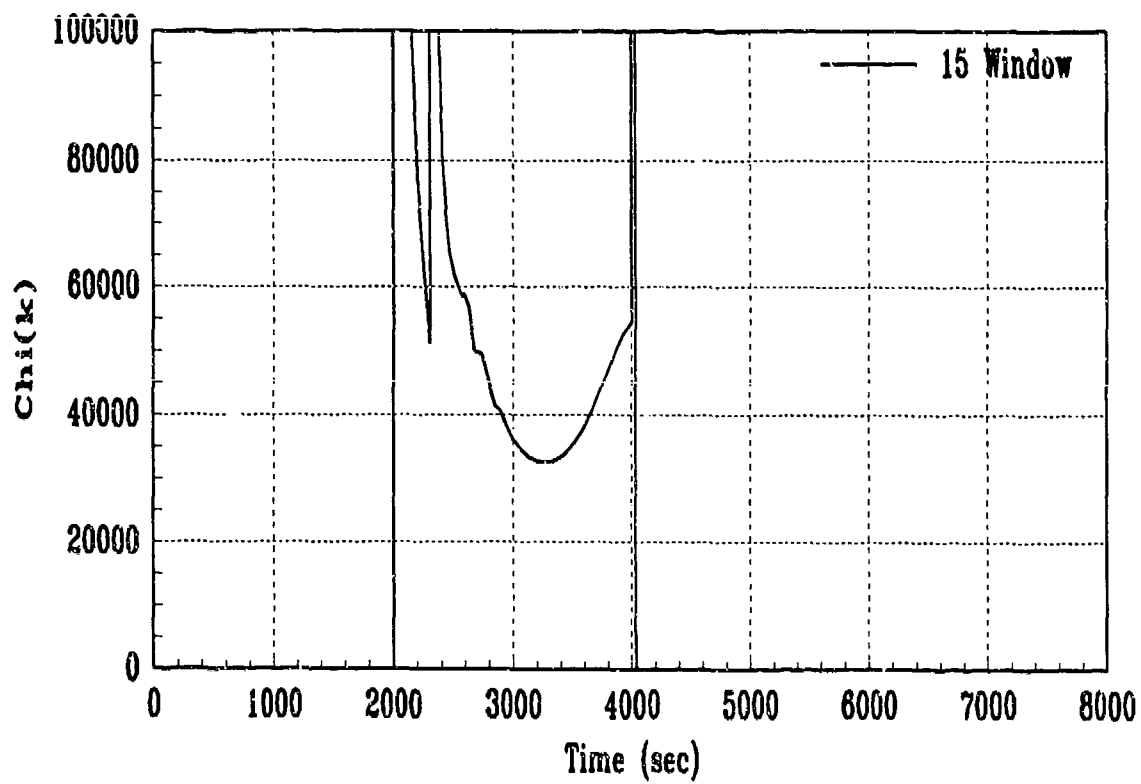
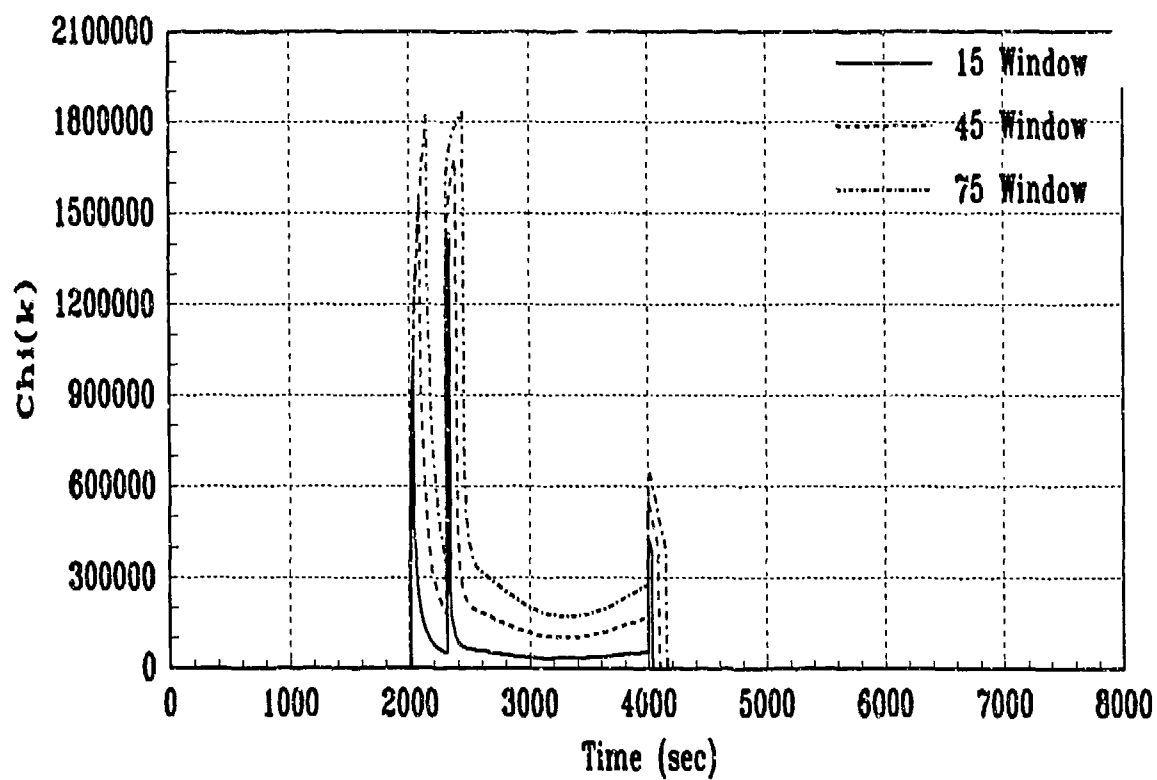


Figure F.9. CHI: Run 4, Bias=7000

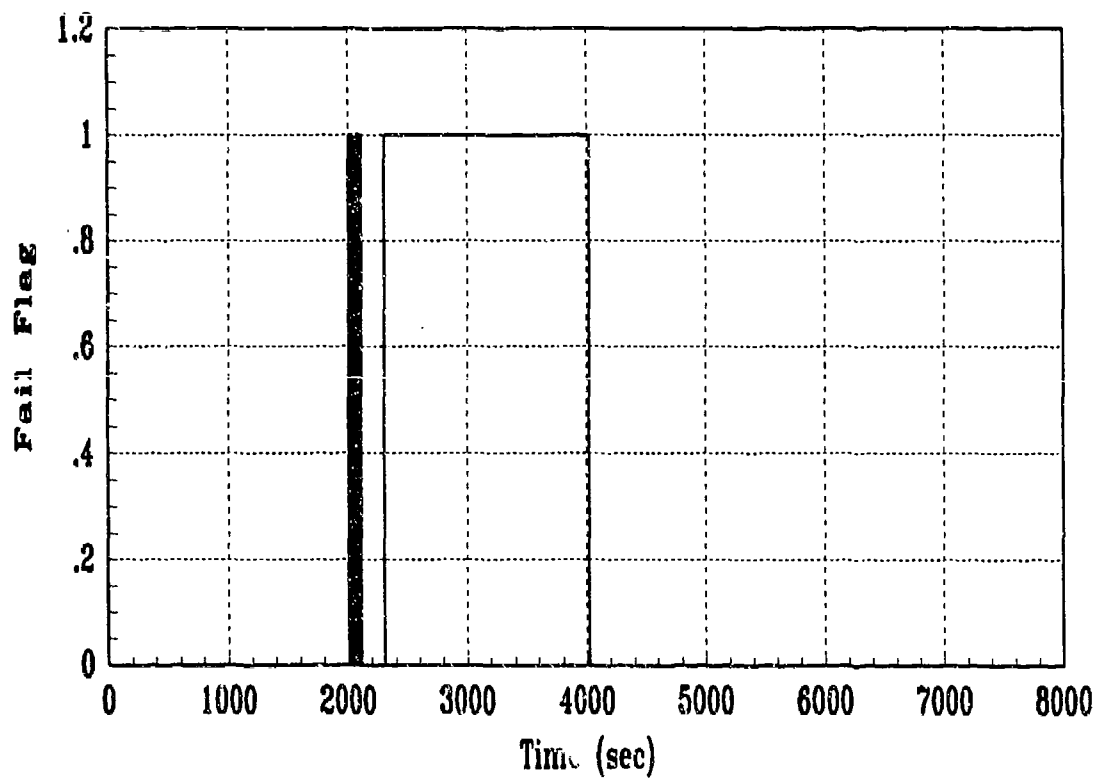
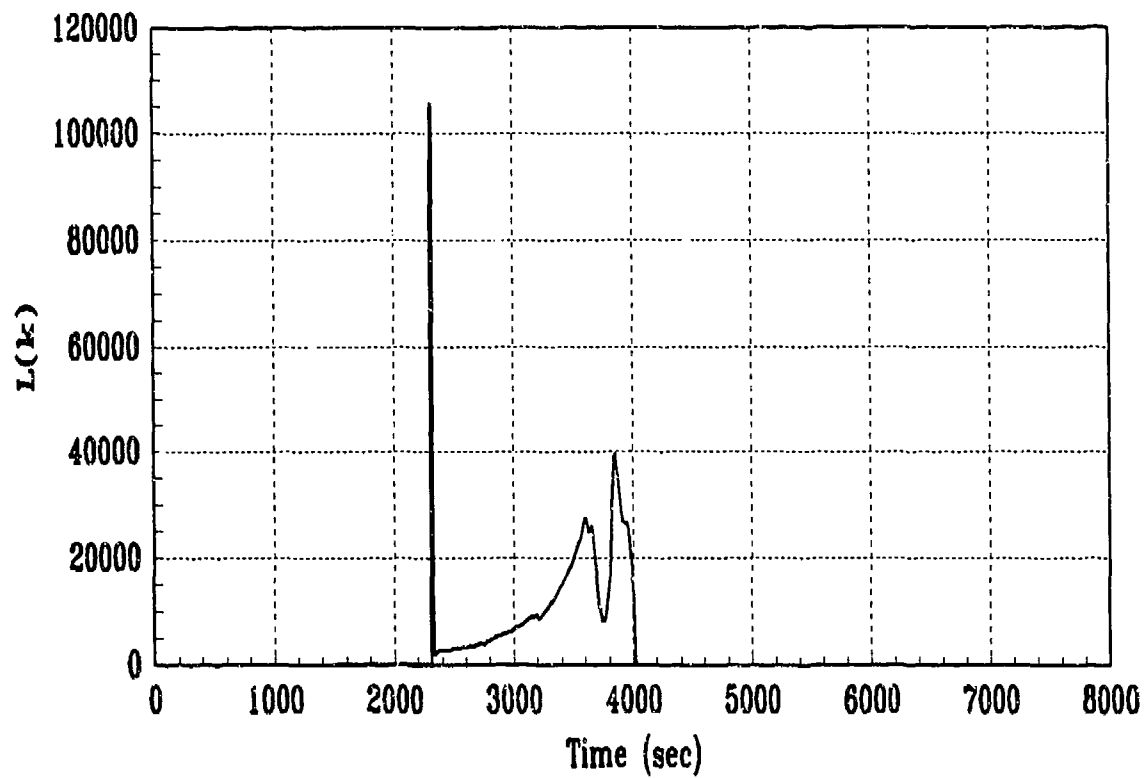


Figure F.10. GLR/Fail Flag: Run 5, Bias=700, 2000-4000 sec

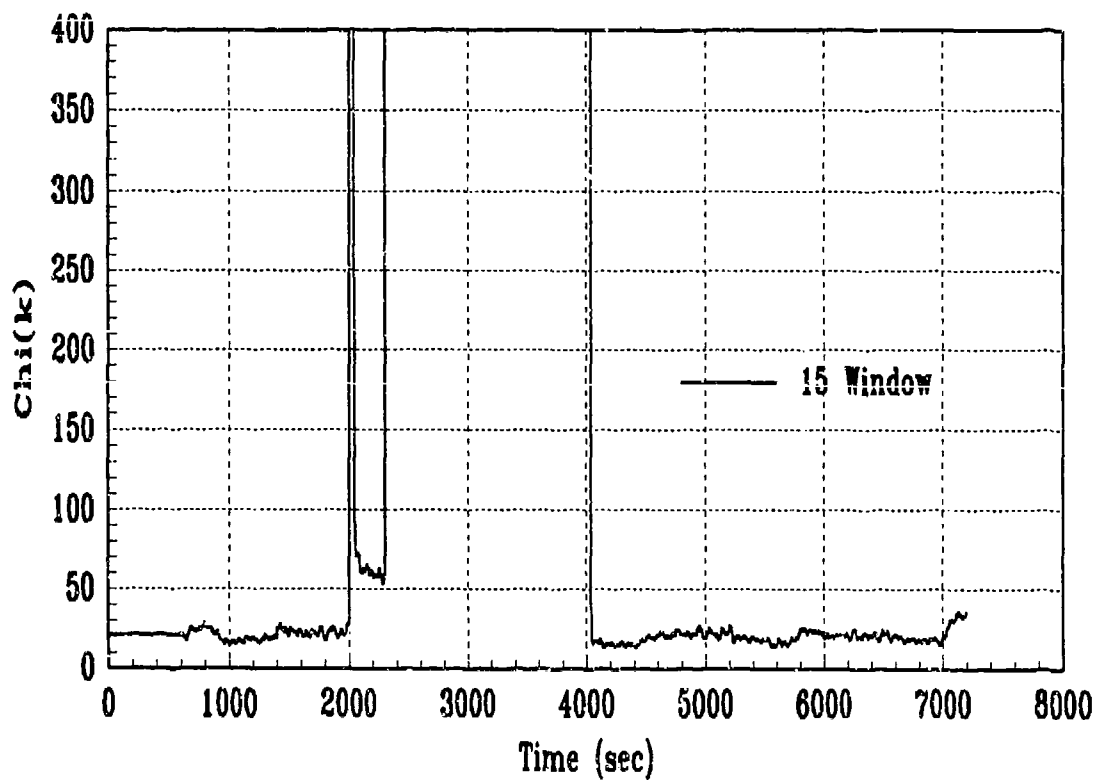
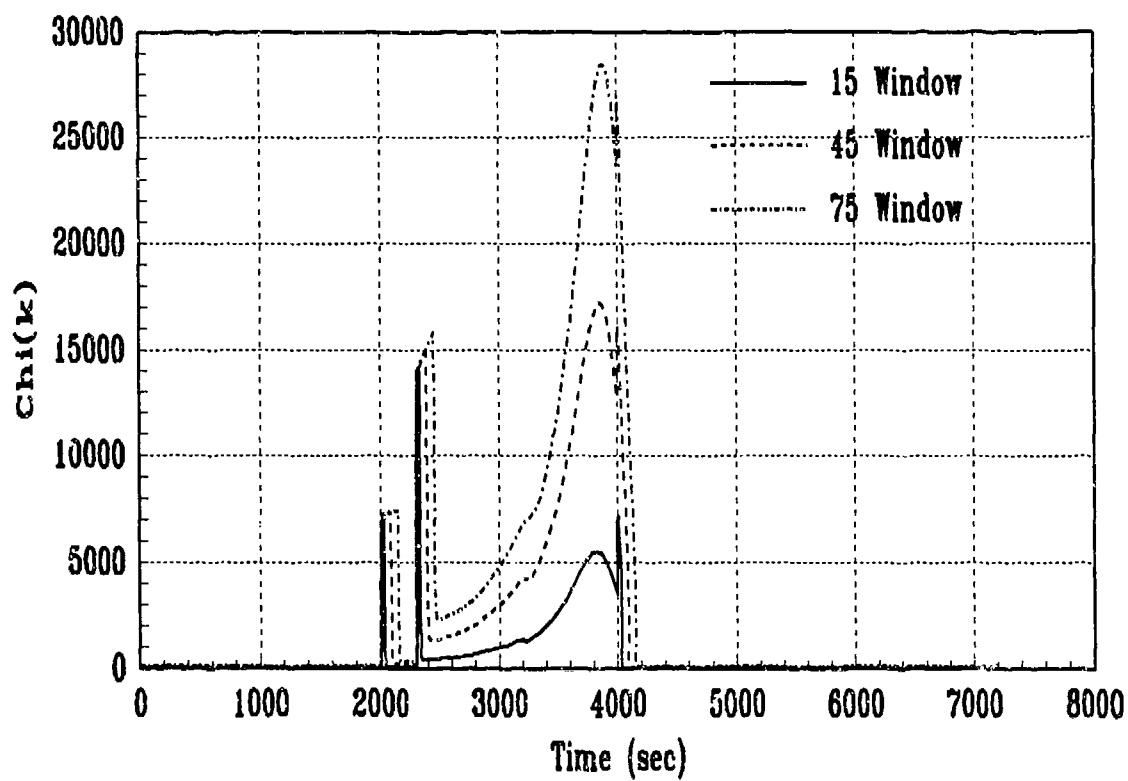


Figure F.11. CHI: Run 5, Bias=700, 2000-4000 sec

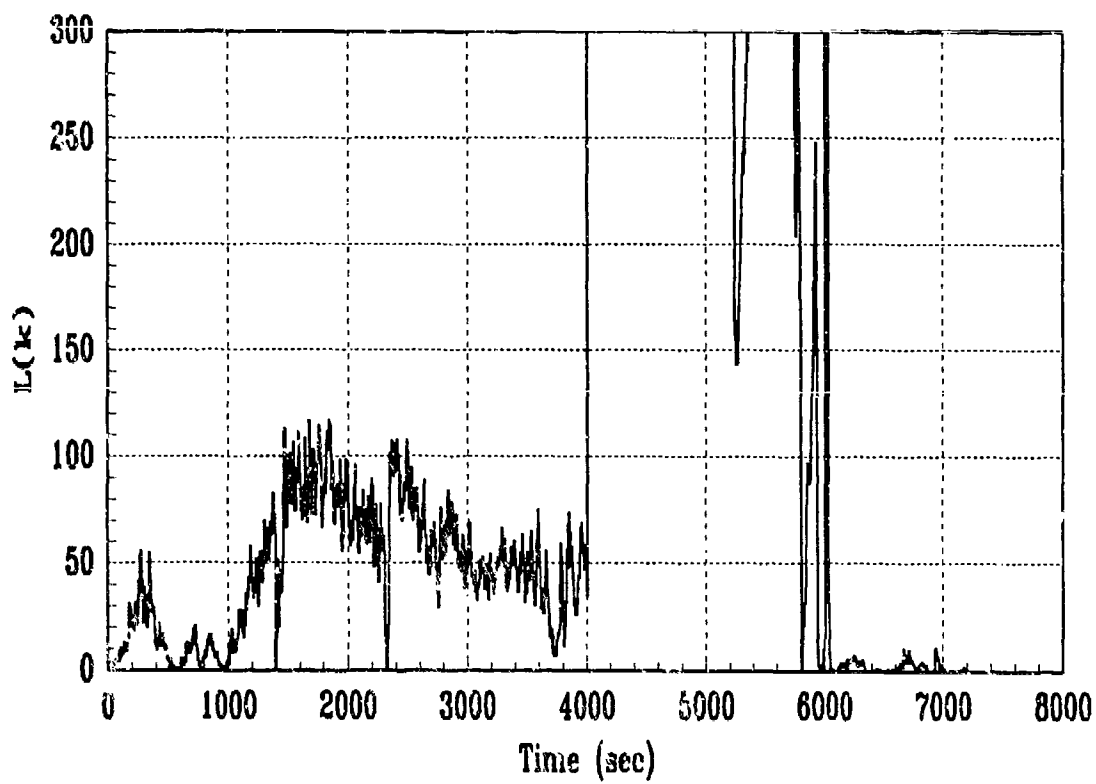
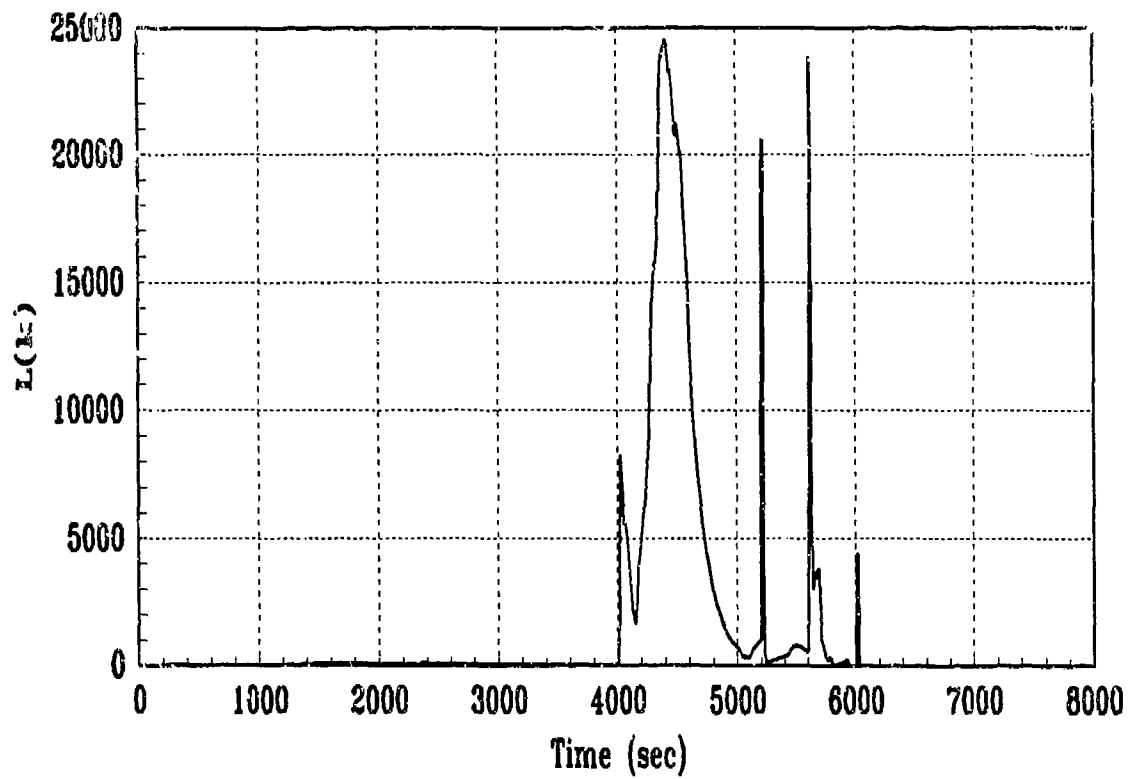
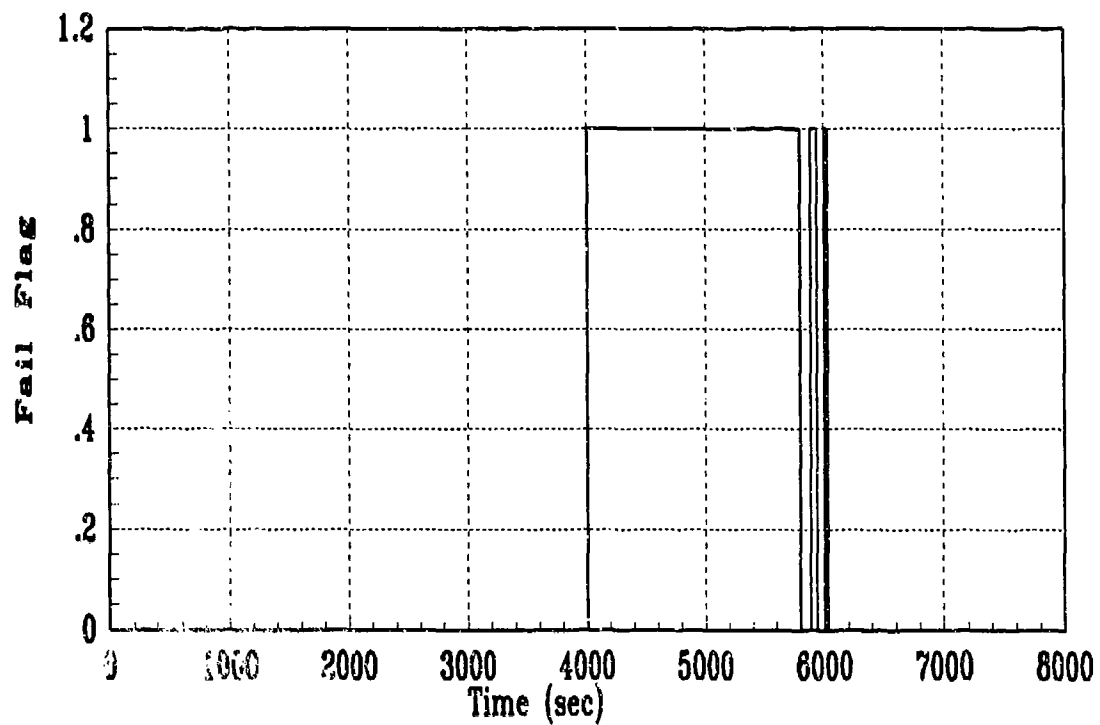
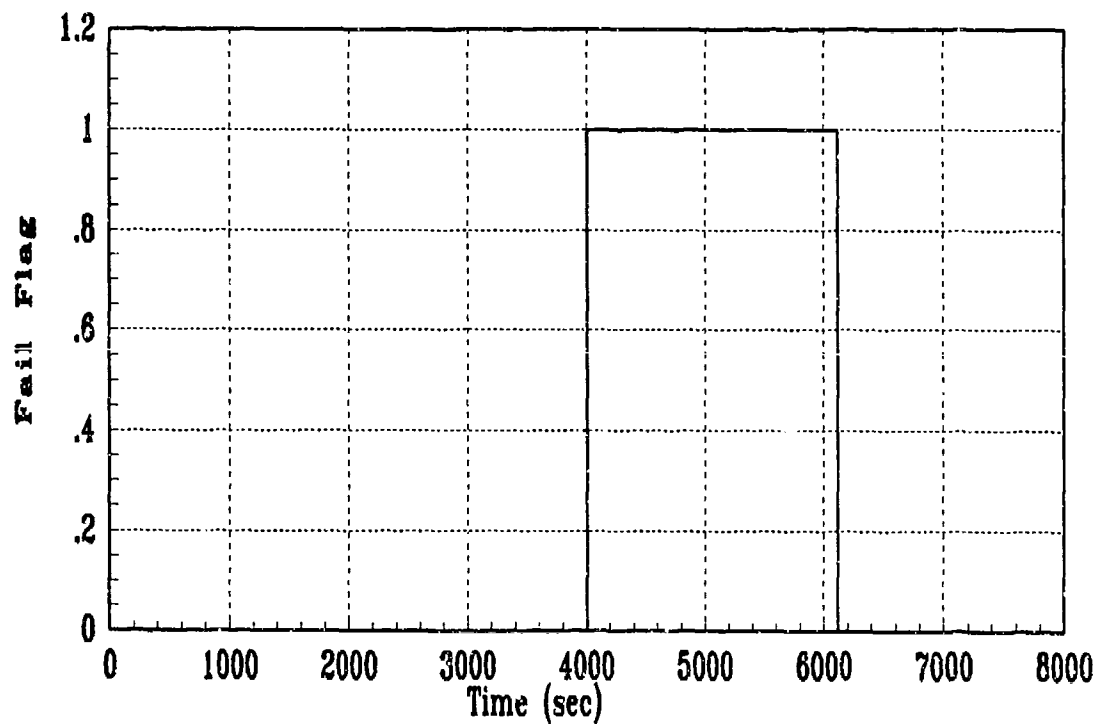


Figure F.12. GLR: Run 6, Bias=700, 4000-6000 sec



Threshold=120 , Number Low=2



Threshold=120 , Number Low=45

Figure F.13. Fail Flag: Run 6, Bias=700, 4000-6000 sec

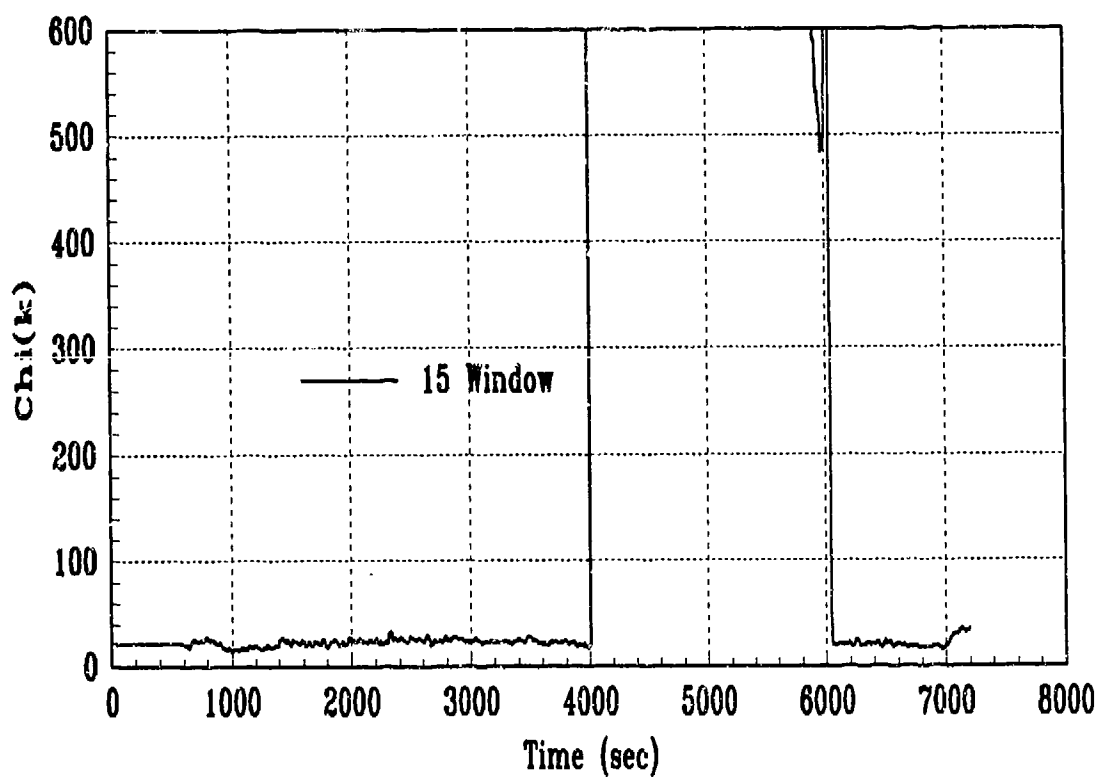
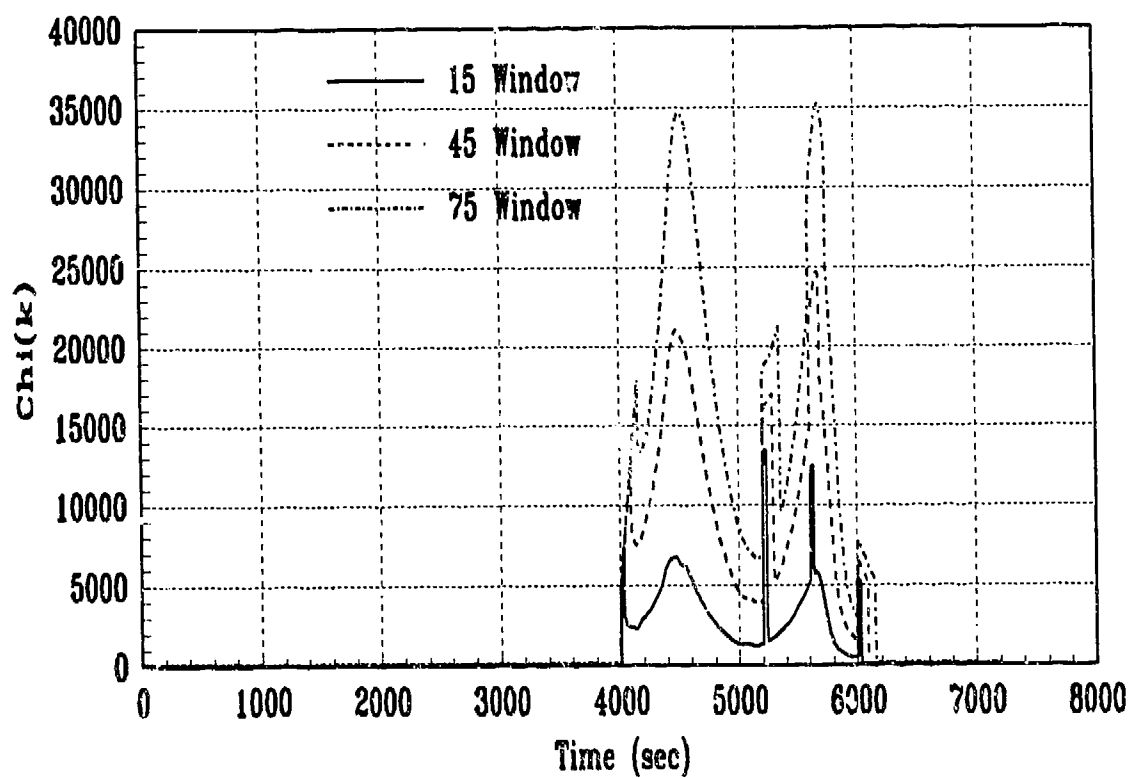


Figure F.14. CHI: Run 6, Bias=700, 4000-6000 sec

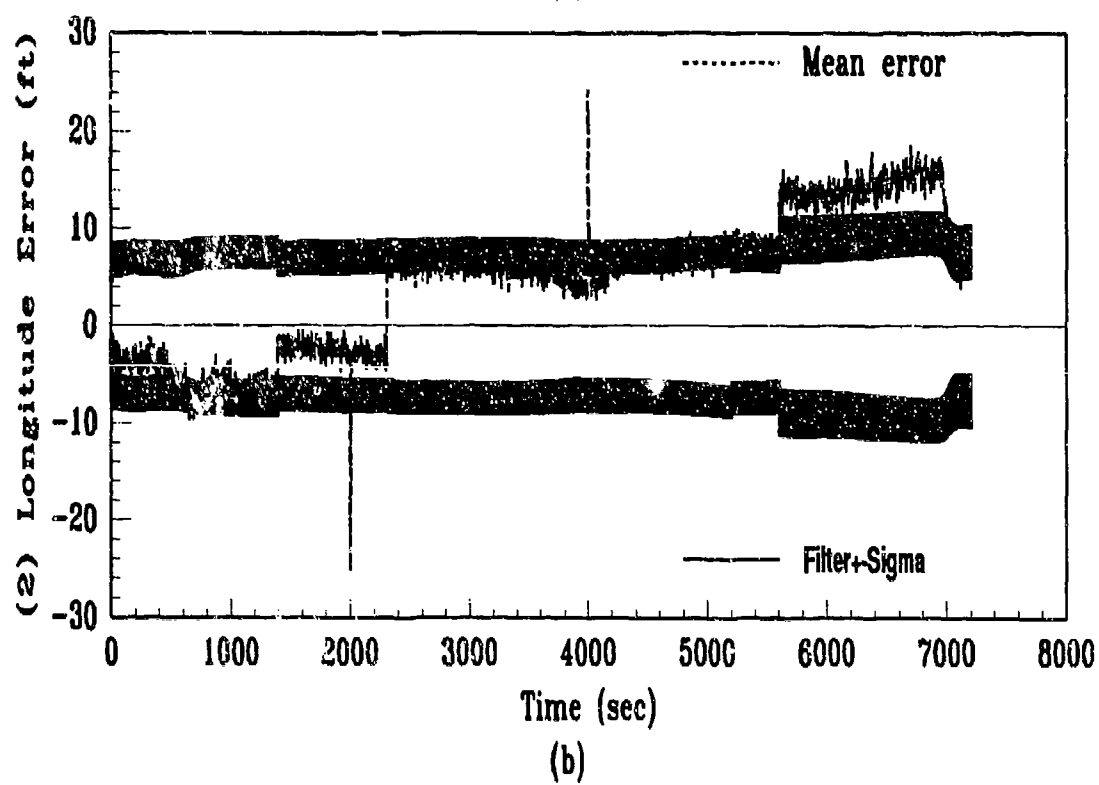
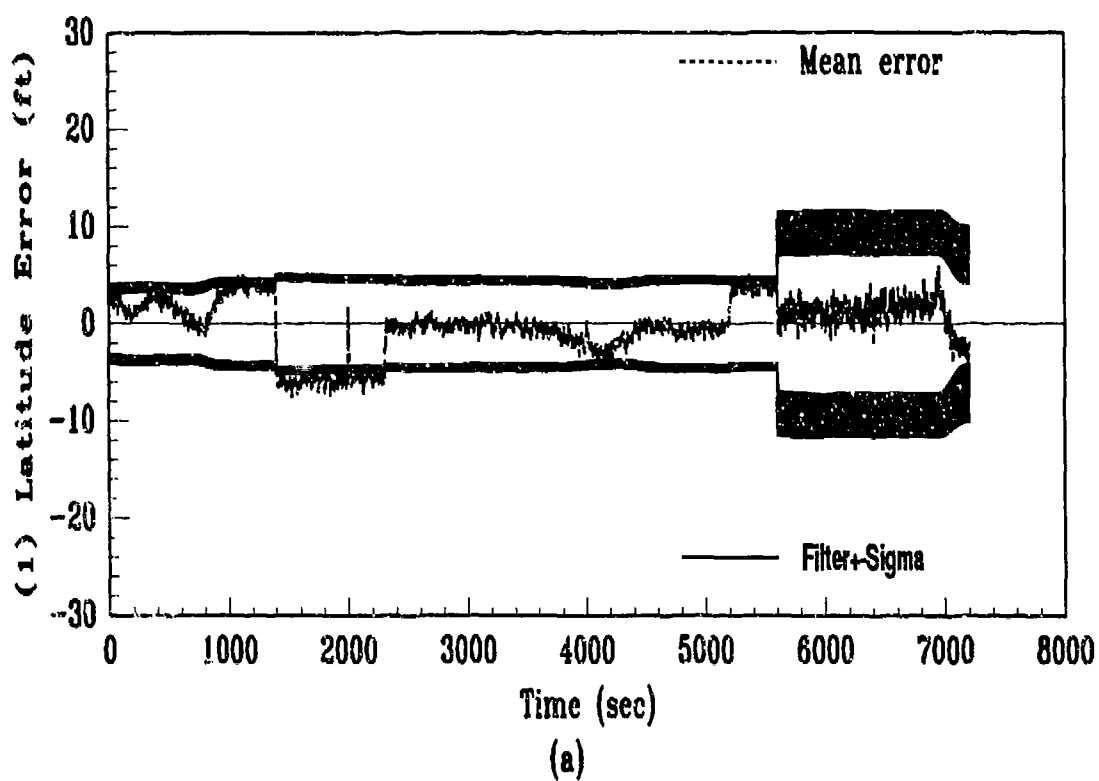
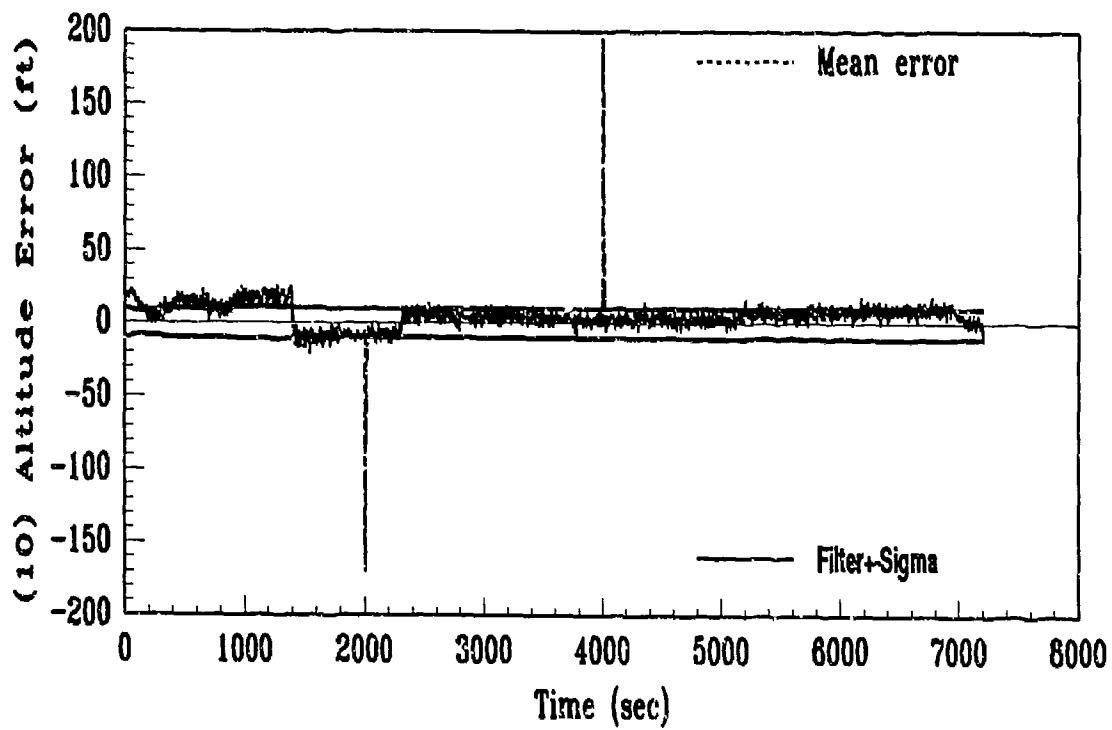
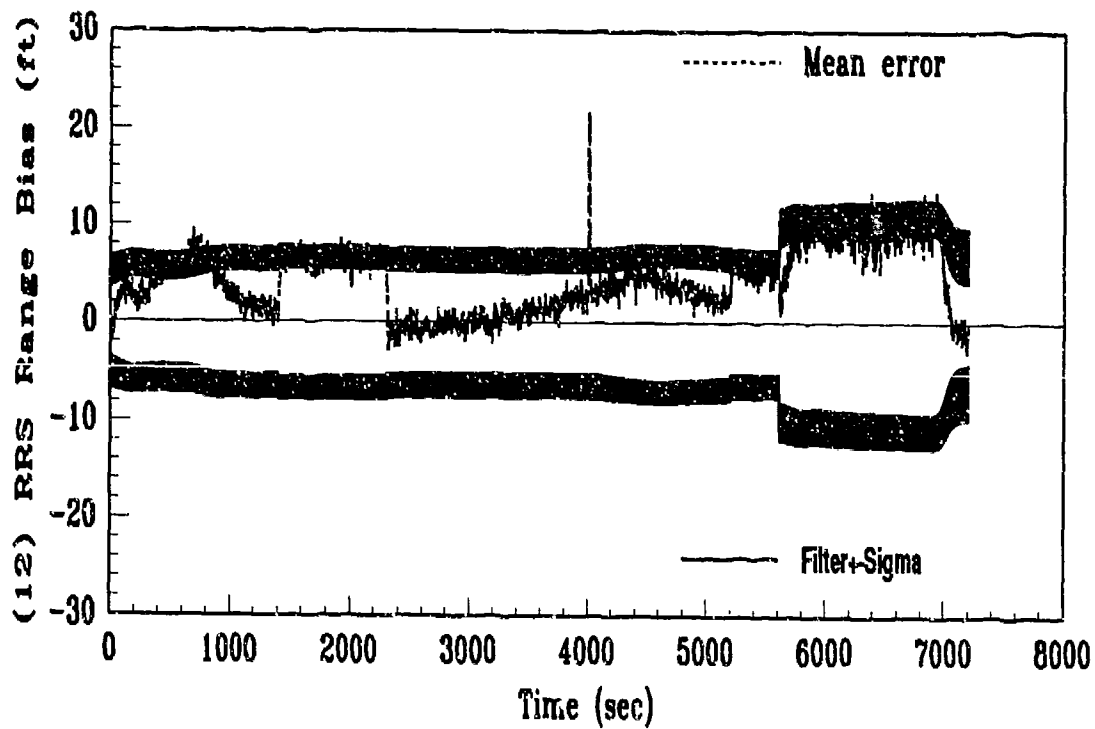


Figure F.15. State Plots: Run 7, Bias=700 on All



(a)



(b)

Figure F.16. State Plots: Run 7, Bias=700 on All

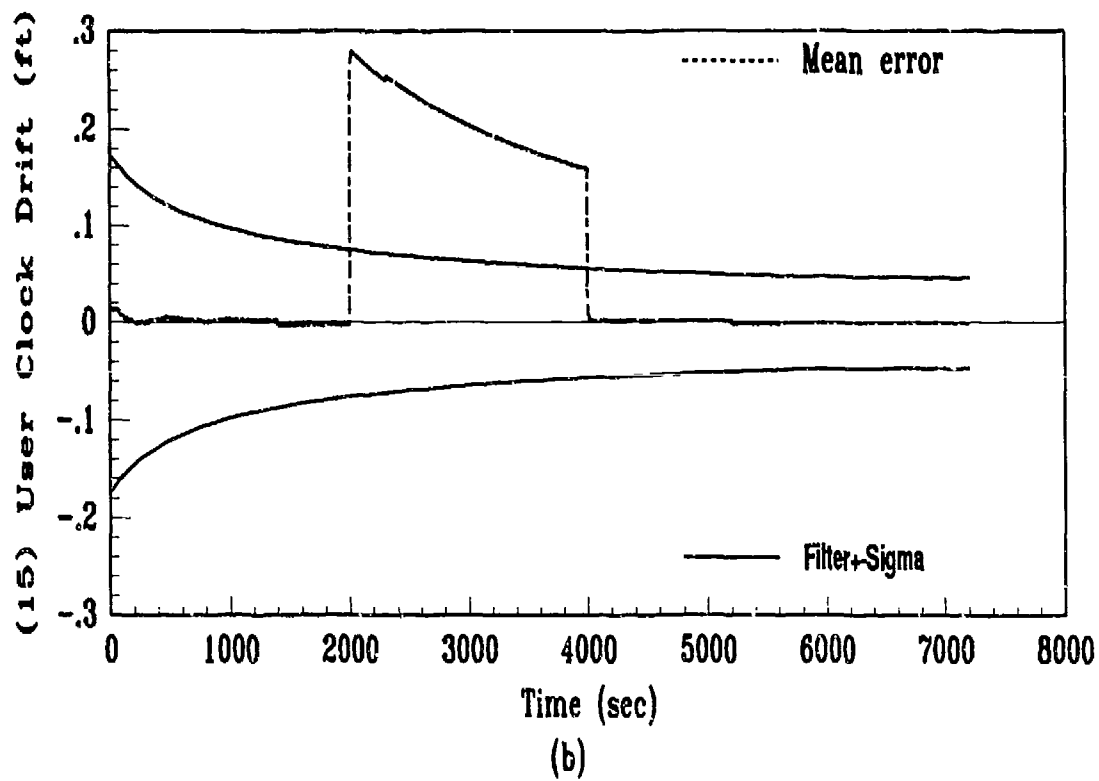
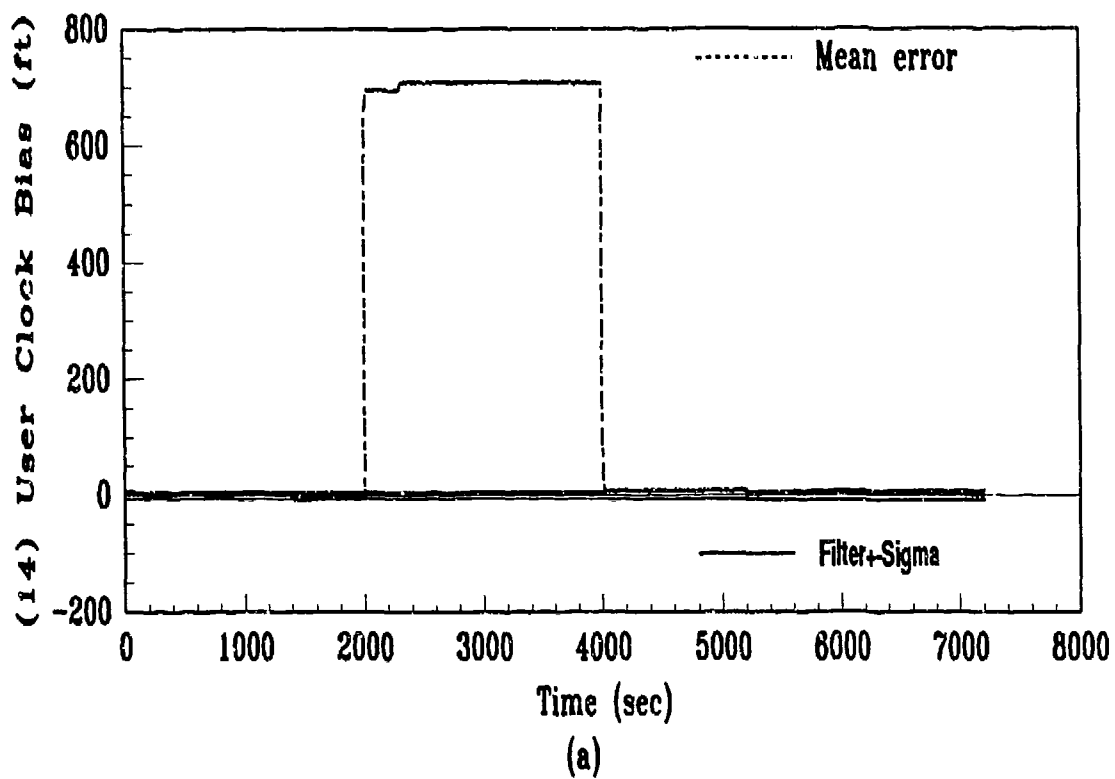


Figure F.17. State Plots: Run 7, Bias=700 on All

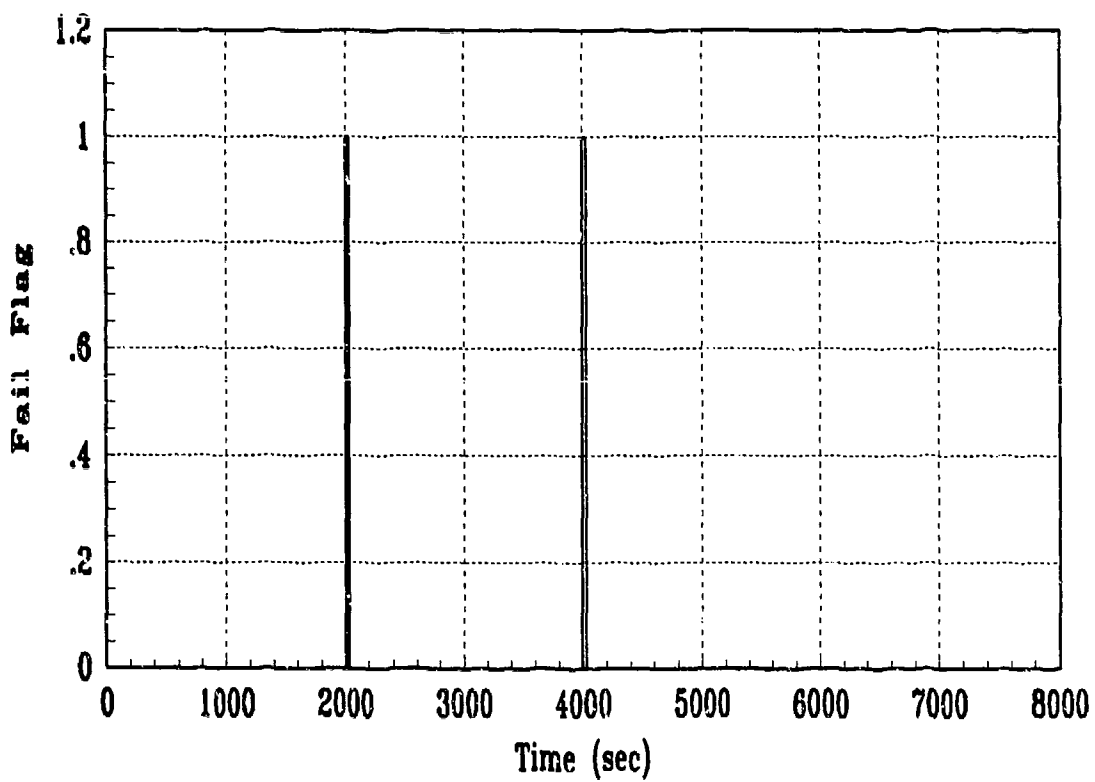
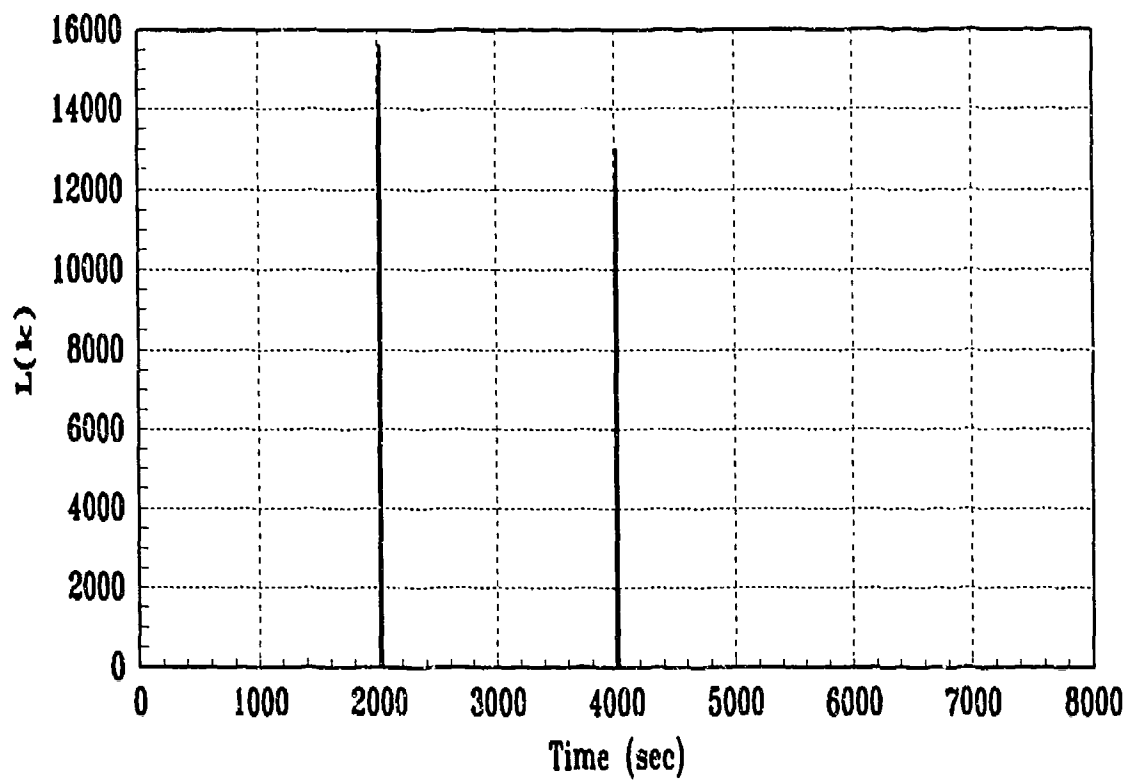
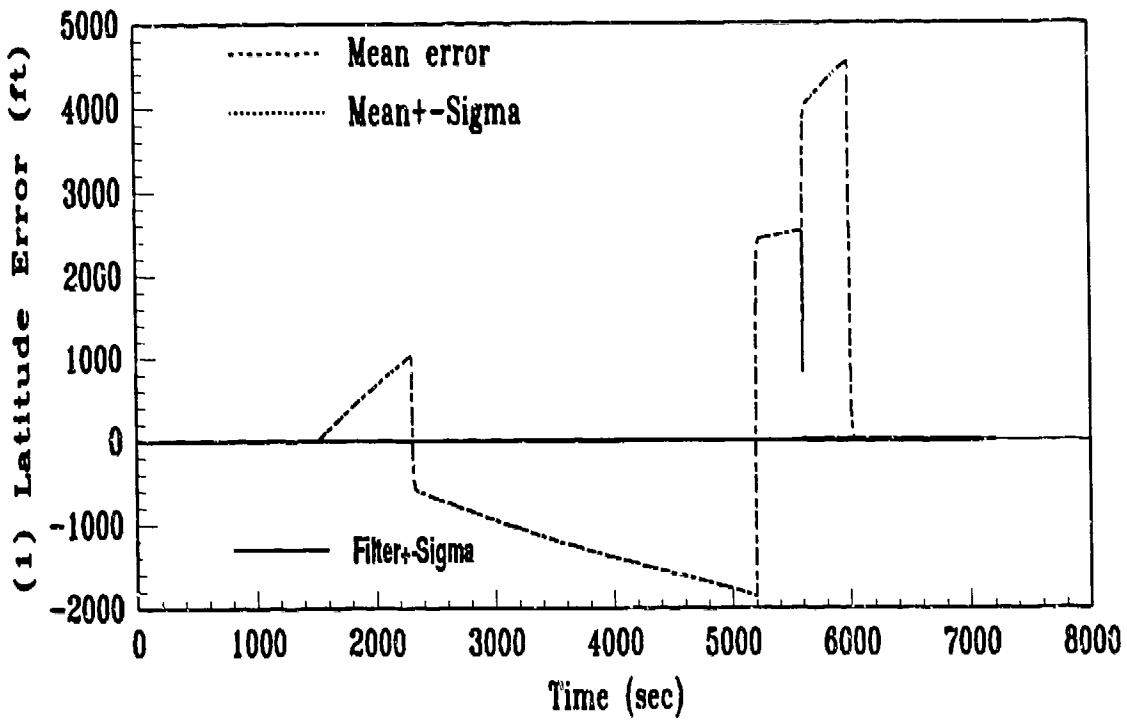


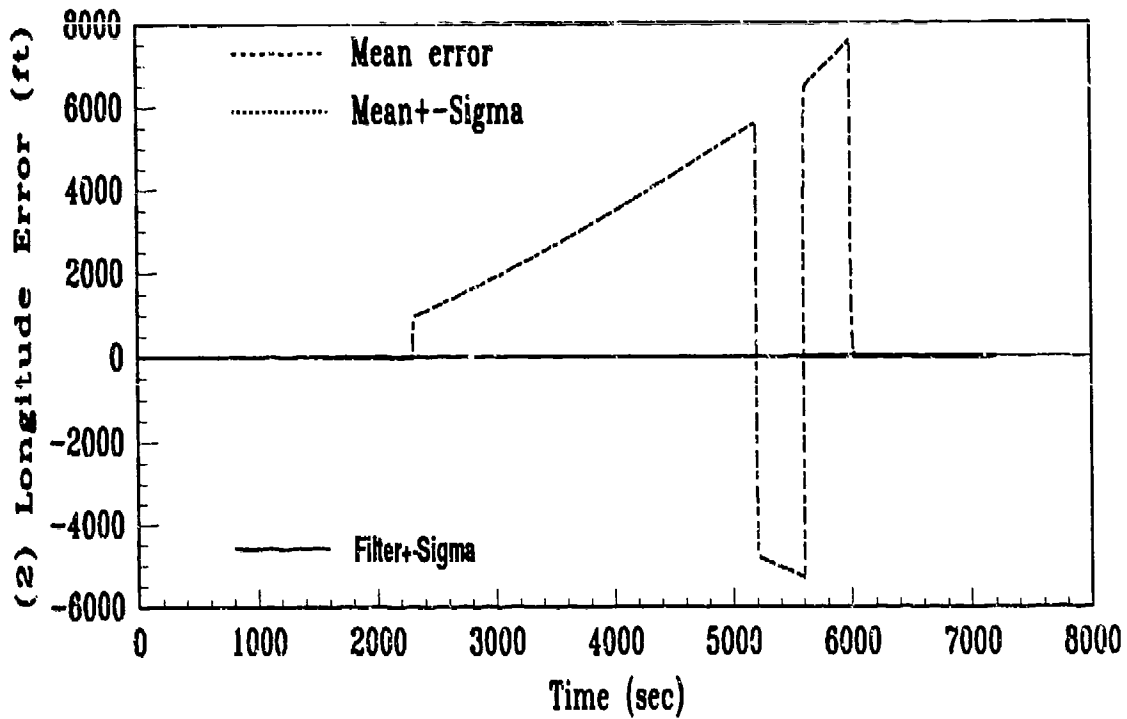
Figure F.18. GLR/Fail Flag: Run 7, Bias=700 on All

Appendix G. *Ramp Failure Plots*

Selected state and residual plots are shown to illustrate the performance of the NRS filter with ramp failures induced on the SV's. Some of the statistics are masked by the large vertical scale of the plots, but this allows the full range of the mean error and residual values to be seen, which best characterizes the filter performance. A variety of GLR and χ plots illustrate the FDI performance for all ramp failures considered. Several plots are shown twice with two different vertical scales to reveal significant details that are lost in other plots.



(a)



(b)

Figure G.1. State Plots: Run 8, Ramp=2T

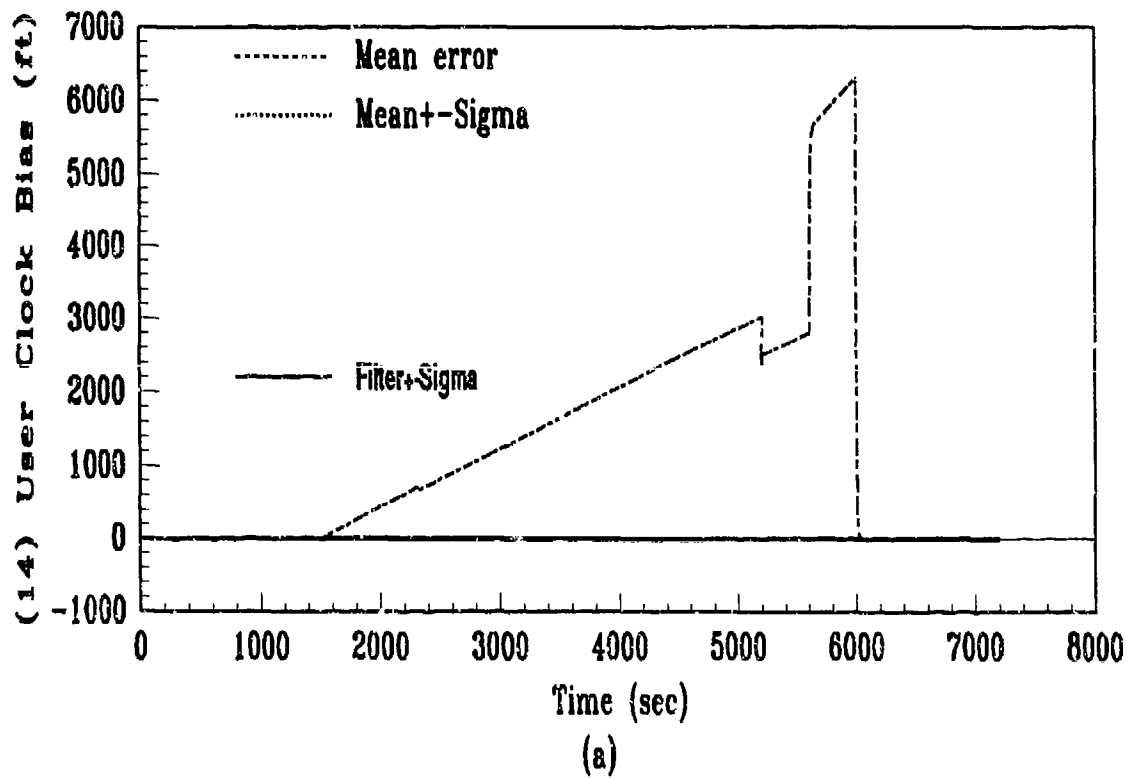
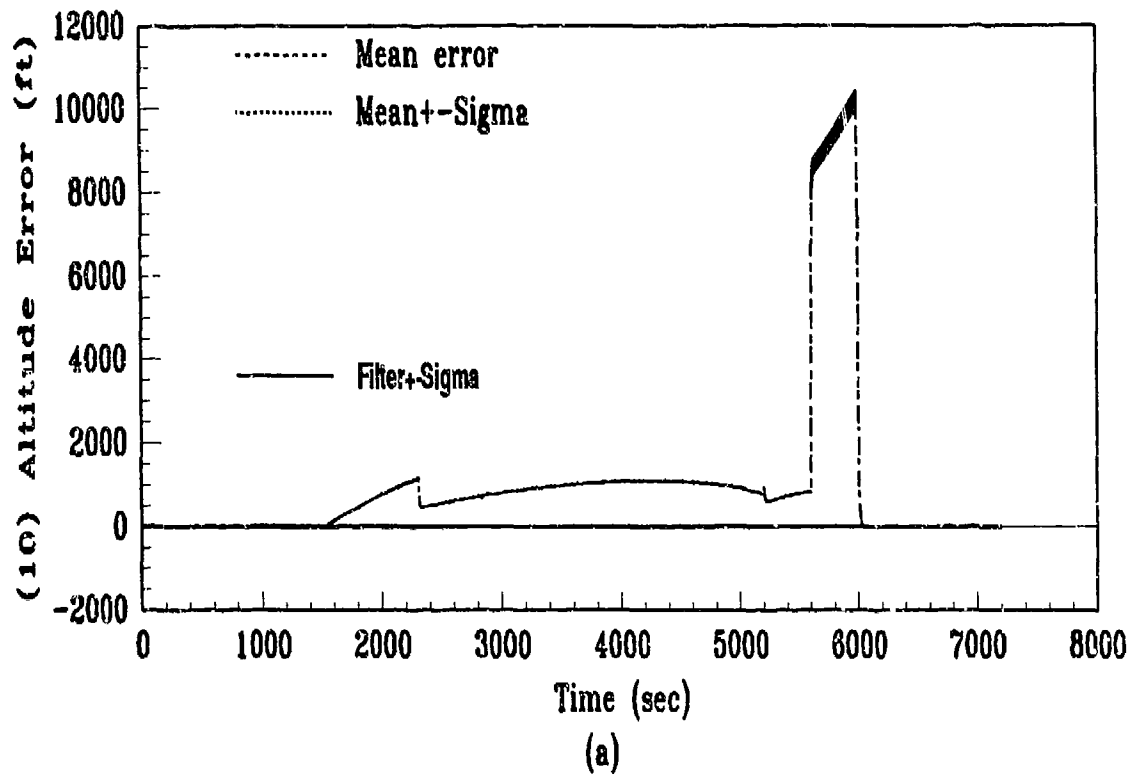
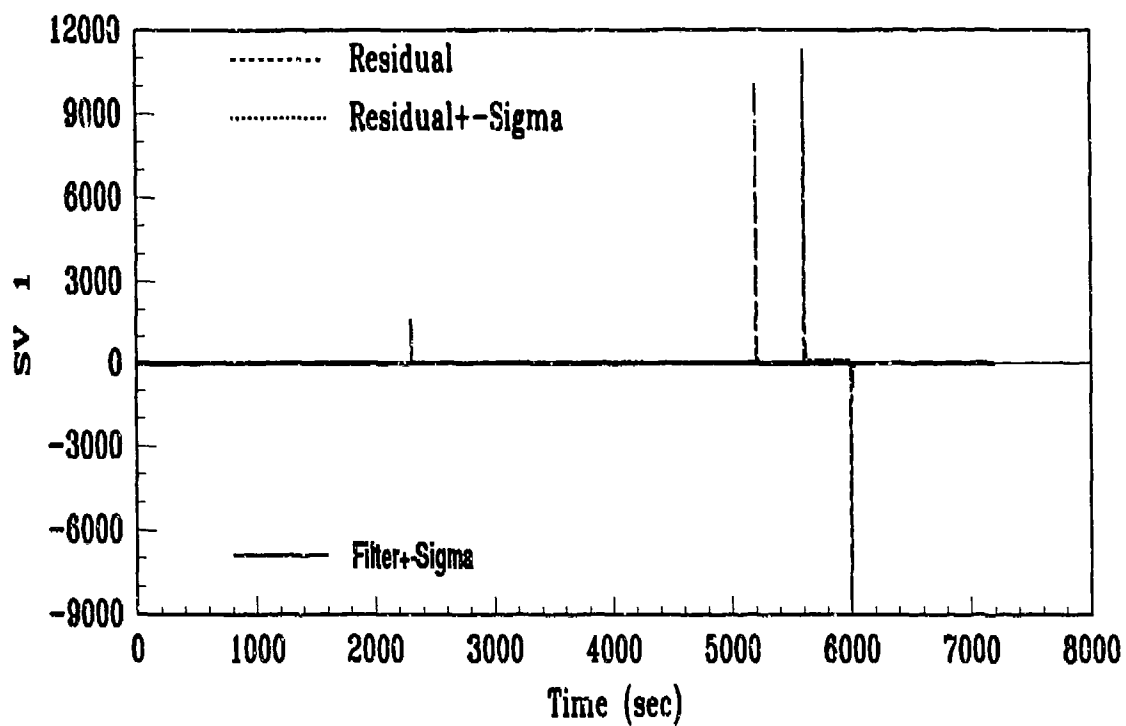
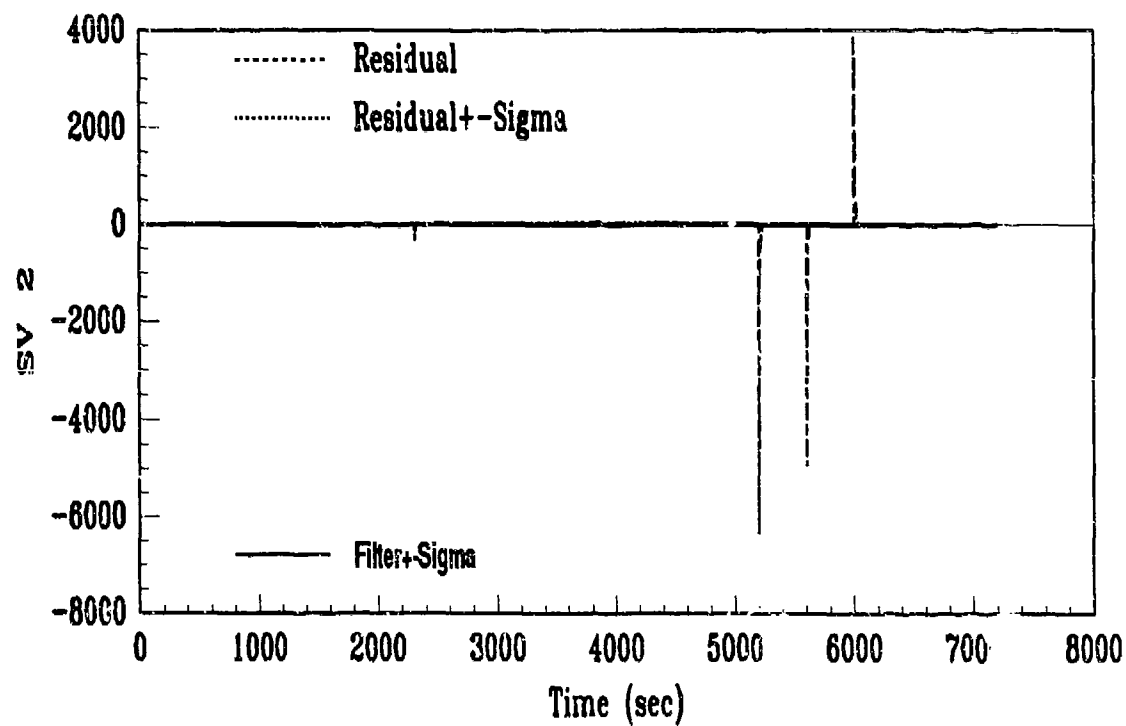


Figure G.2. State Plots: Run 8, Ramp=2T



(a)



(b)

Figure G.3. Residual Plots: Run 8, Ramp=2T

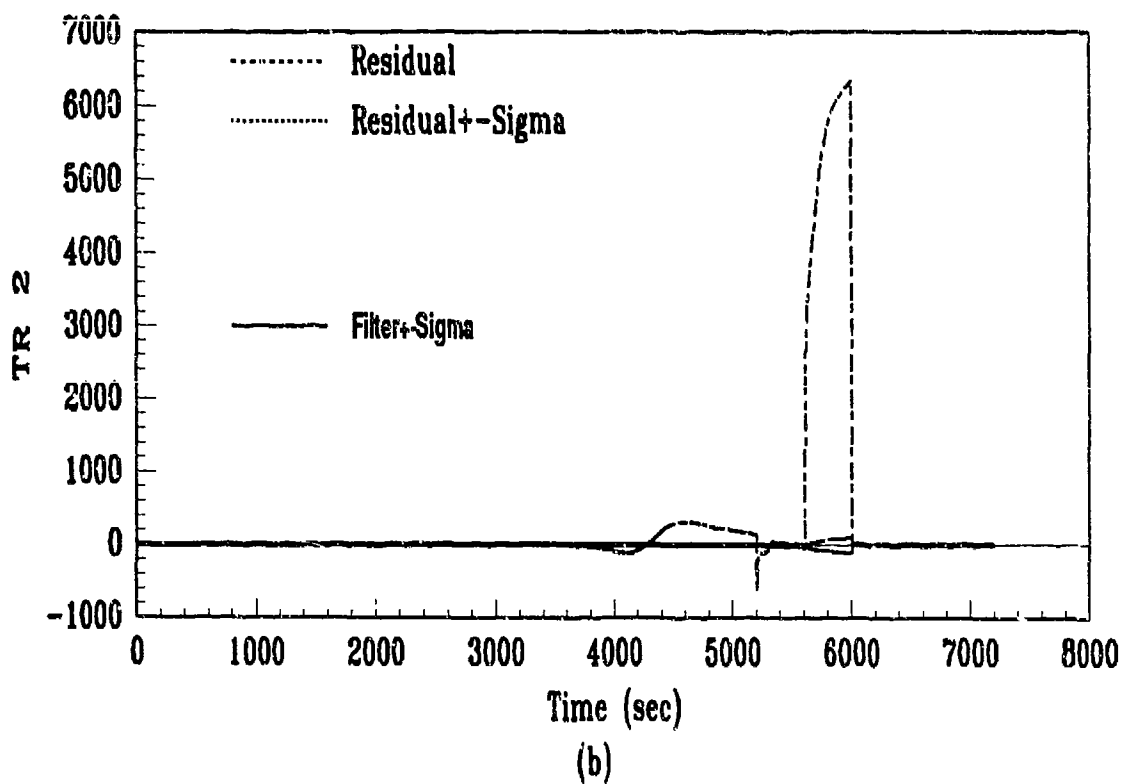
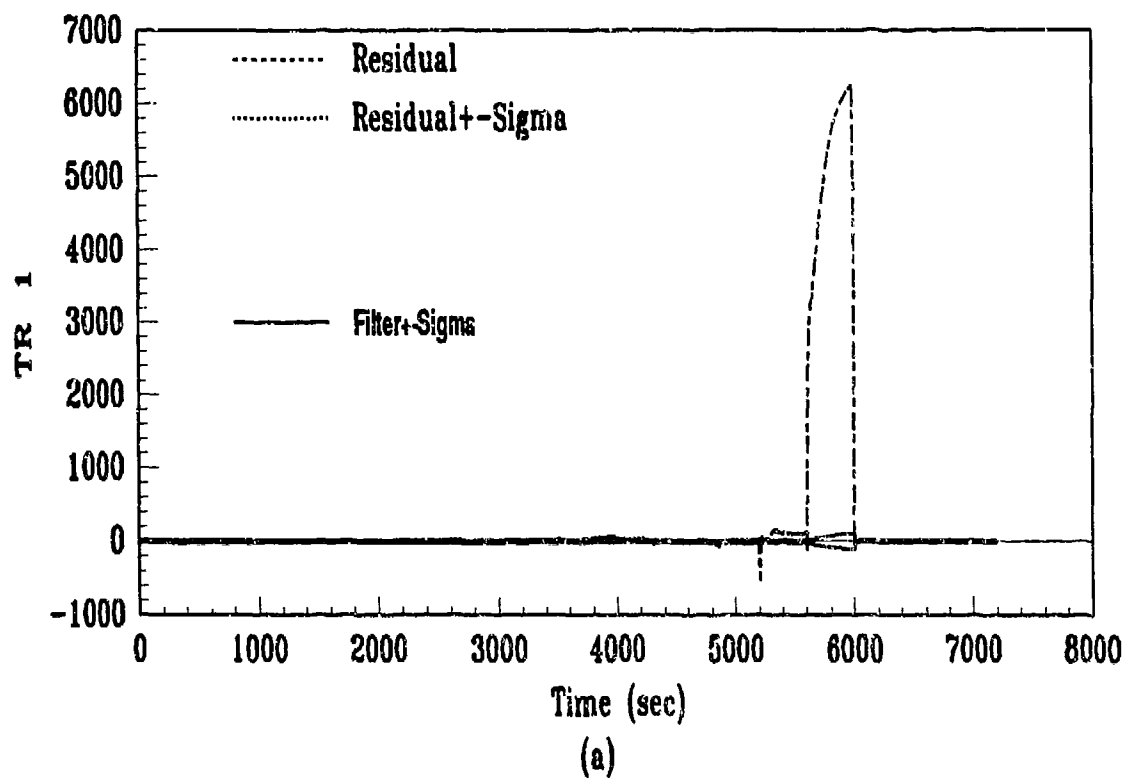


Figure G.4. Residual Plots: Run 8, Ramp=2T

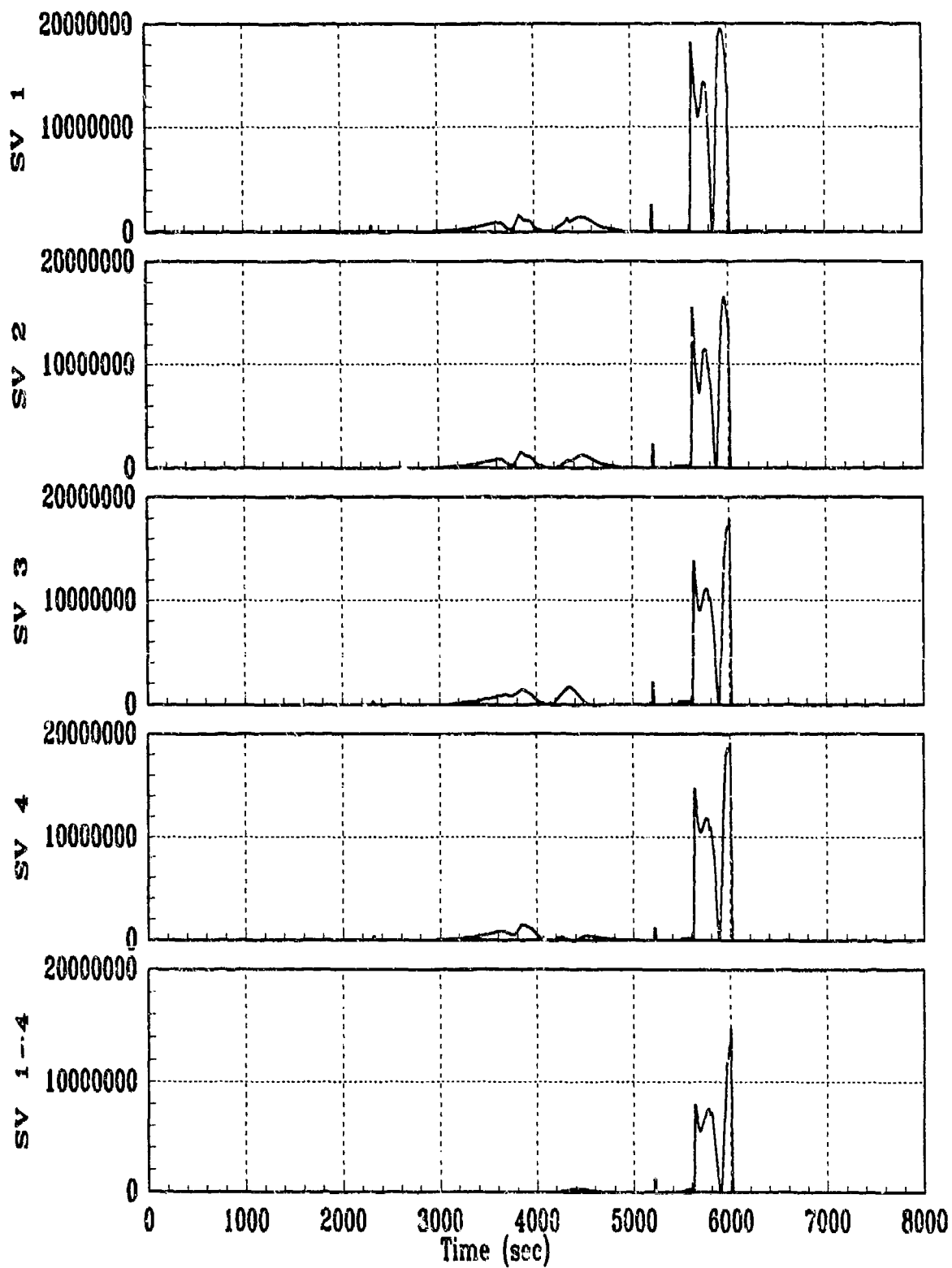


Figure G.5. GLR: Run 8, Ramp=2T

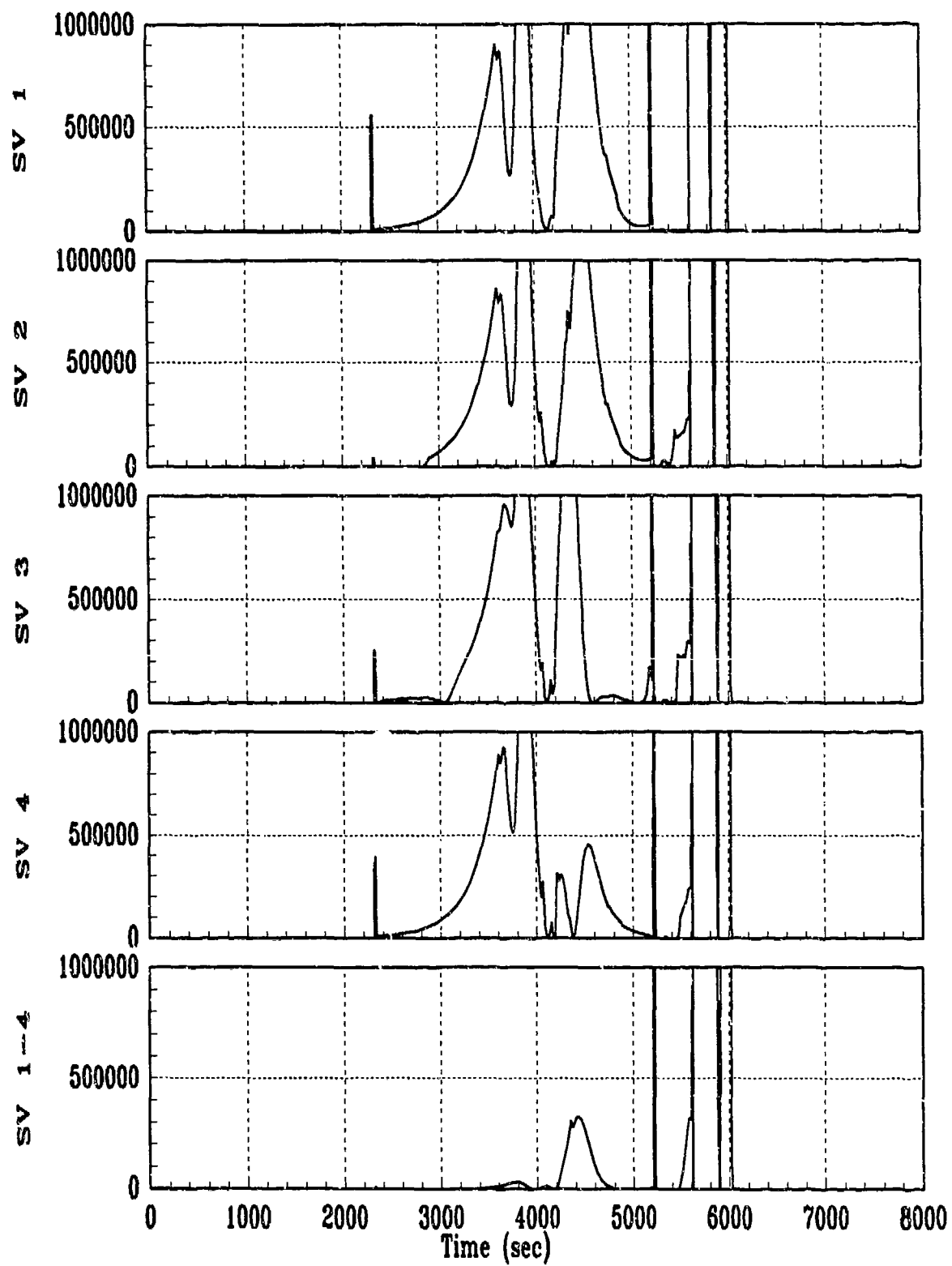
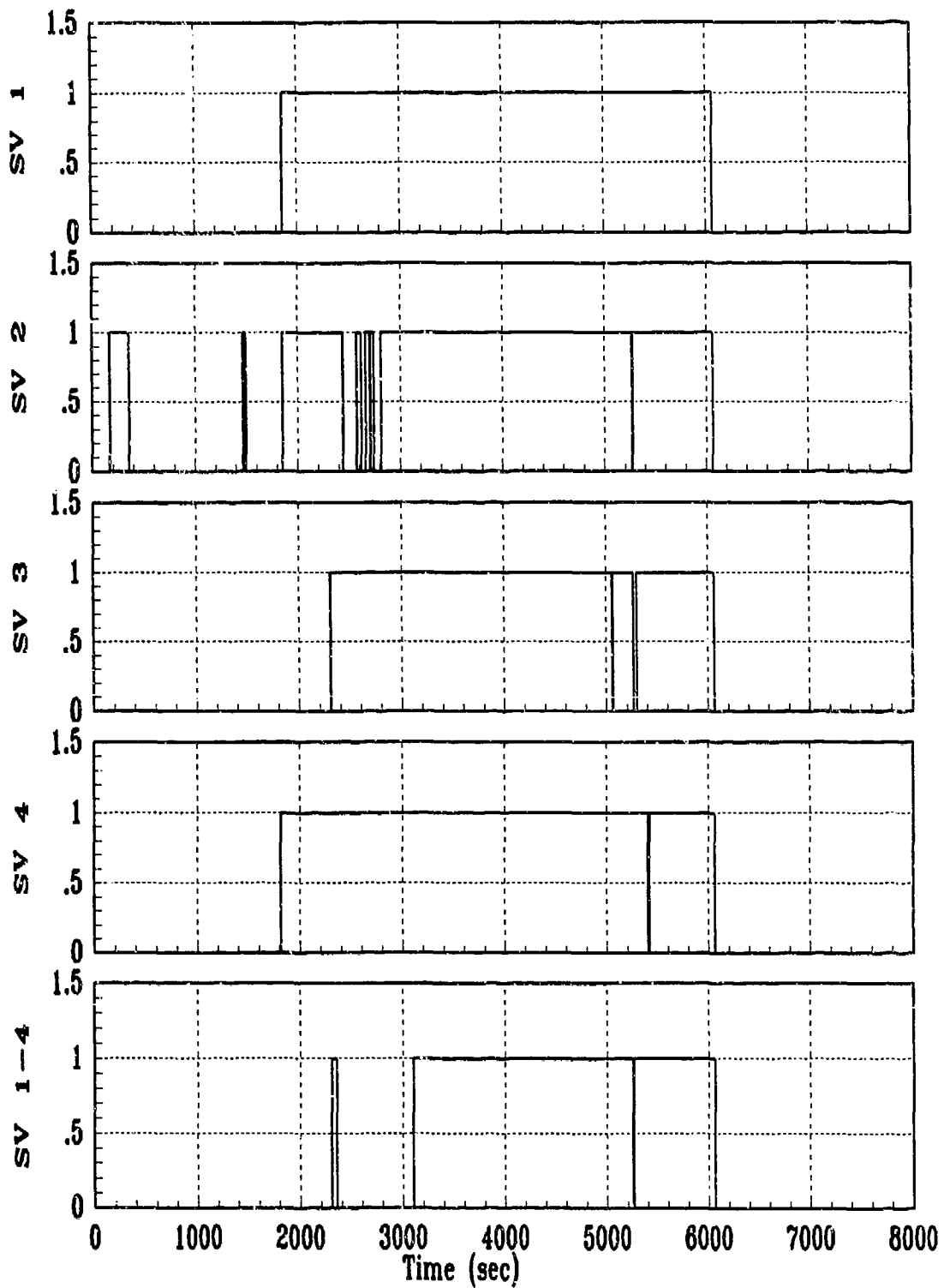


Figure G.6. GLR: Run 8, Ramp=2T



Threshold=120 , Number Low=10

Figure G.7. Fail Flag: Run 8, Ramp=2T

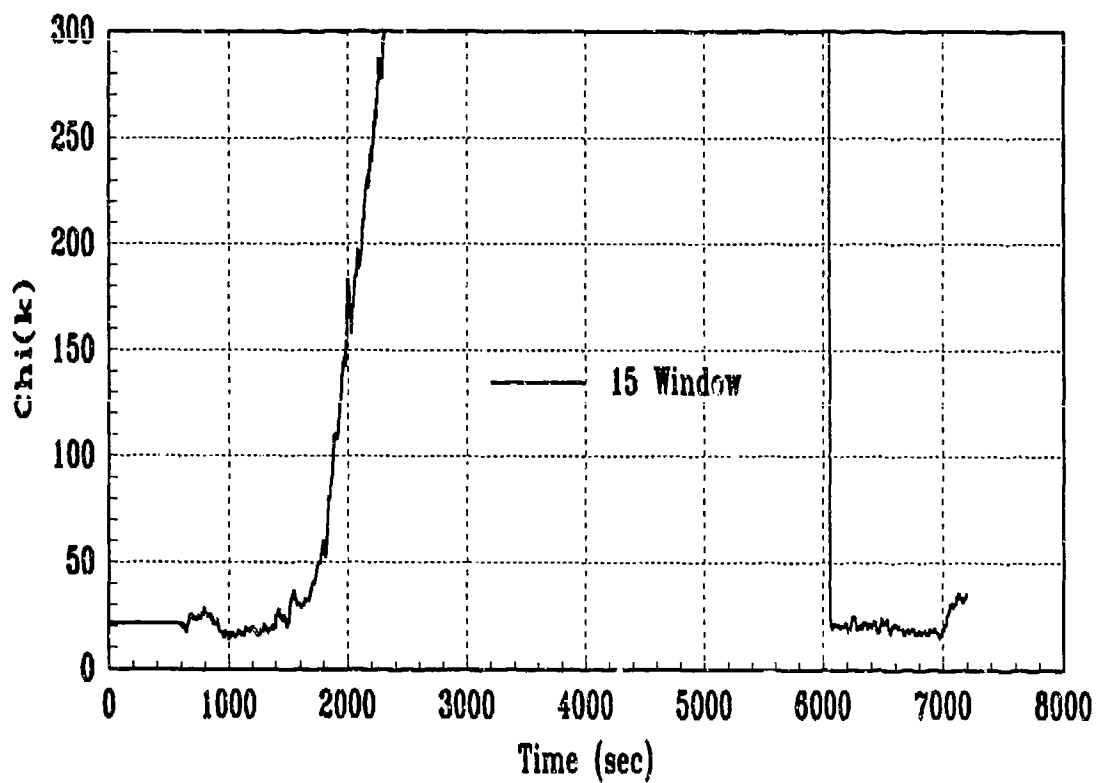
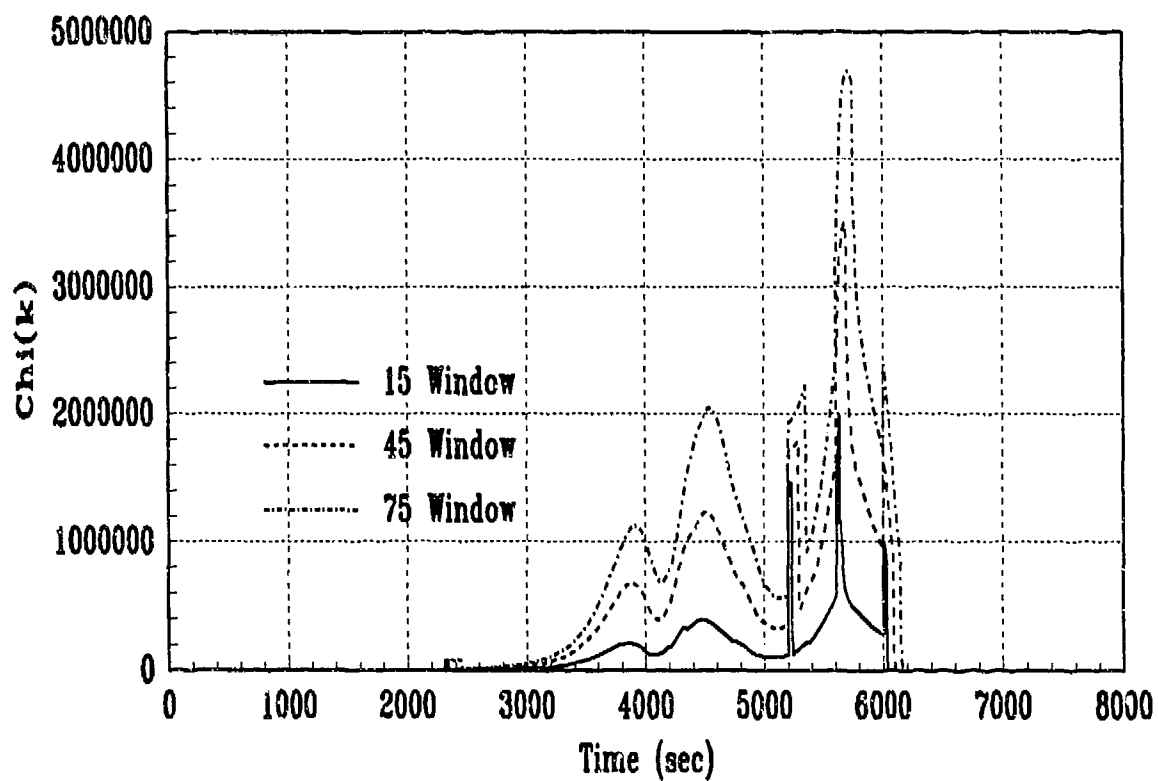


Figure G.8. CHI: Run 8, Ramp=2T

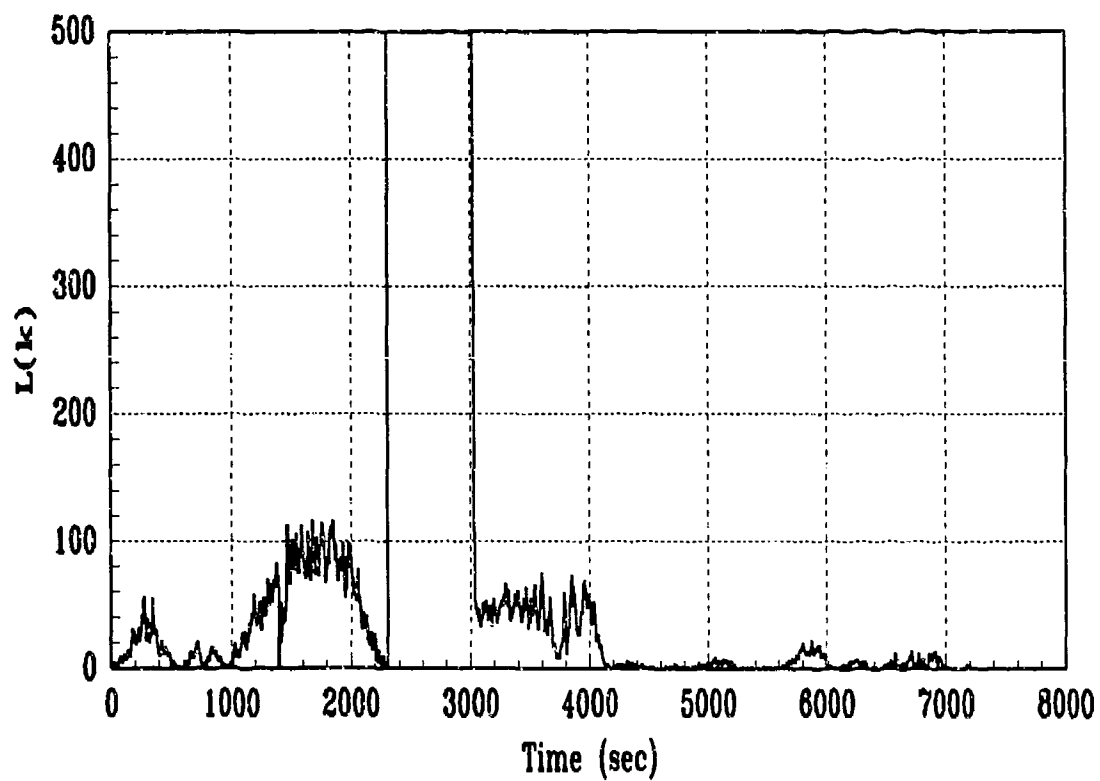
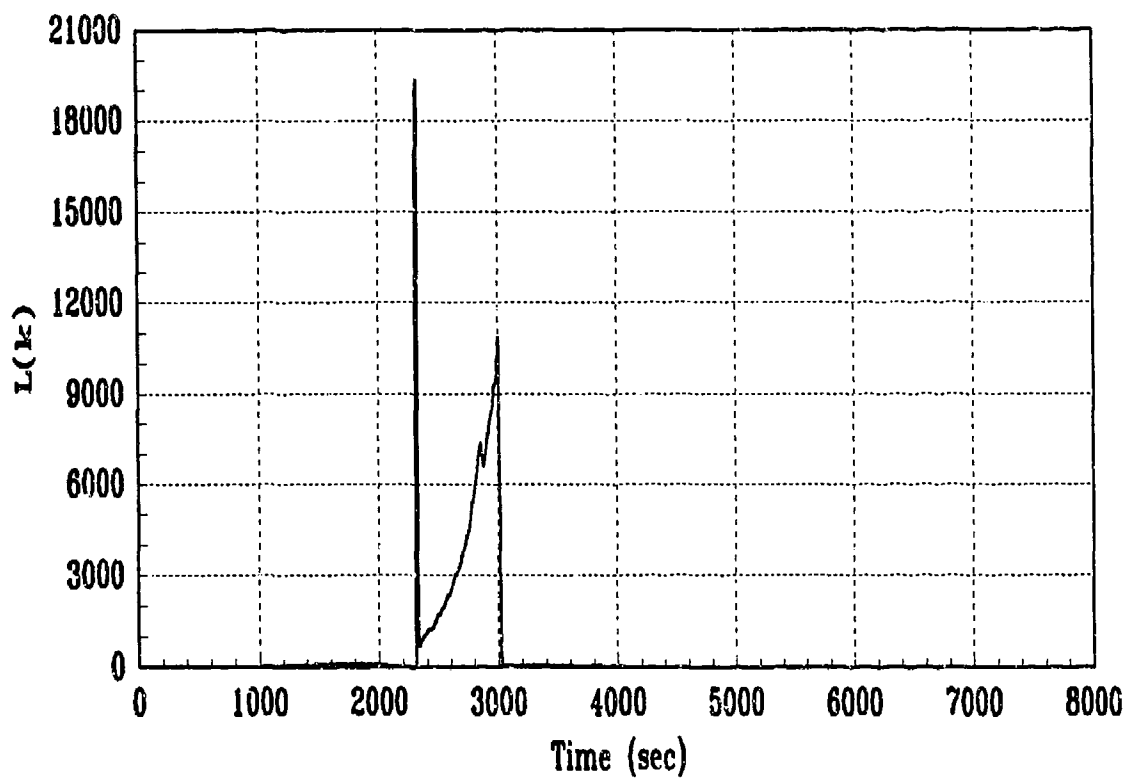
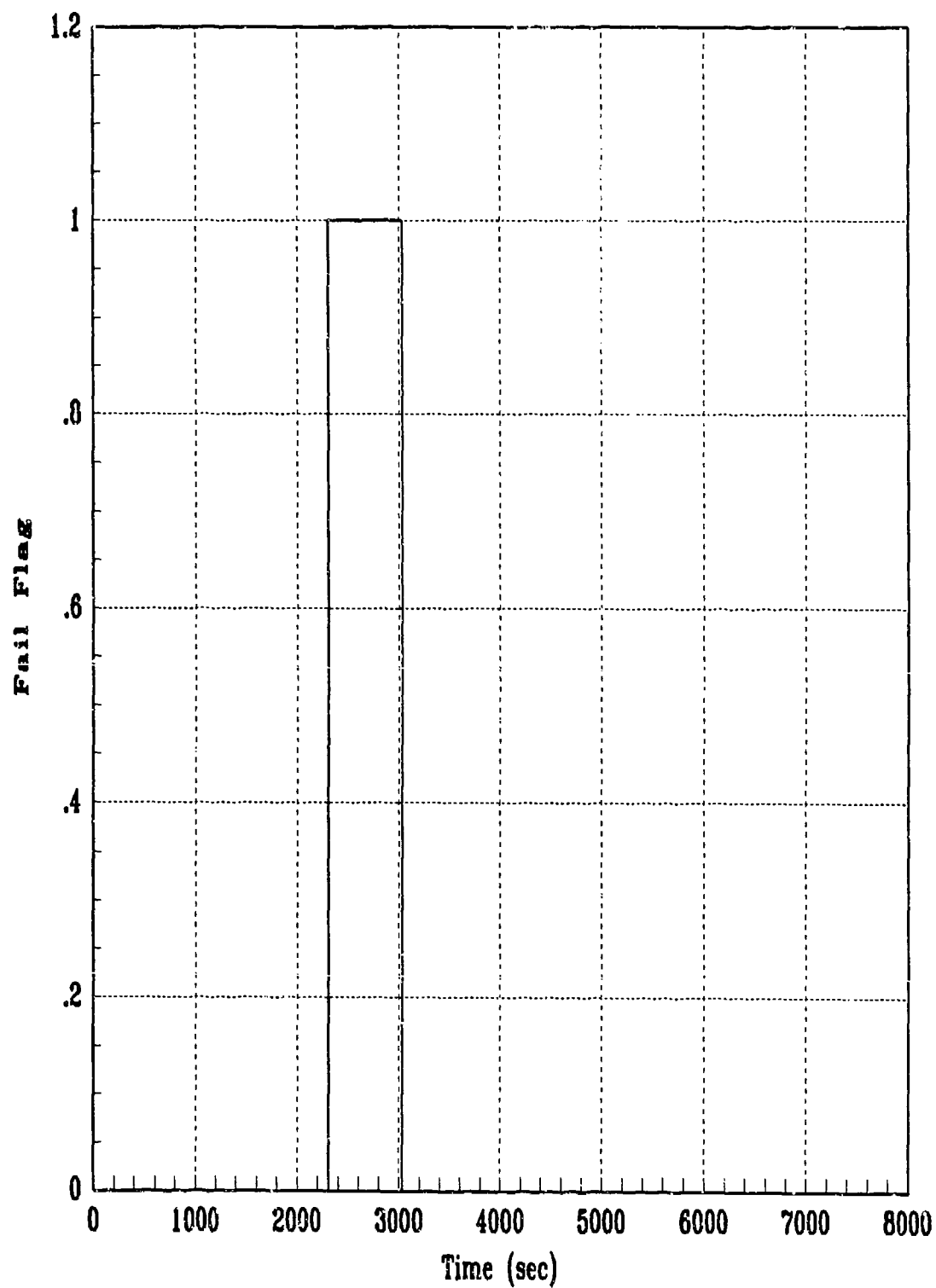


Figure G.9. GLR: Run 9, Ramp=1T



Threshold=120 , Number Low=2

Figure G.10. Fail Flag: Run 9, Ramp=1T

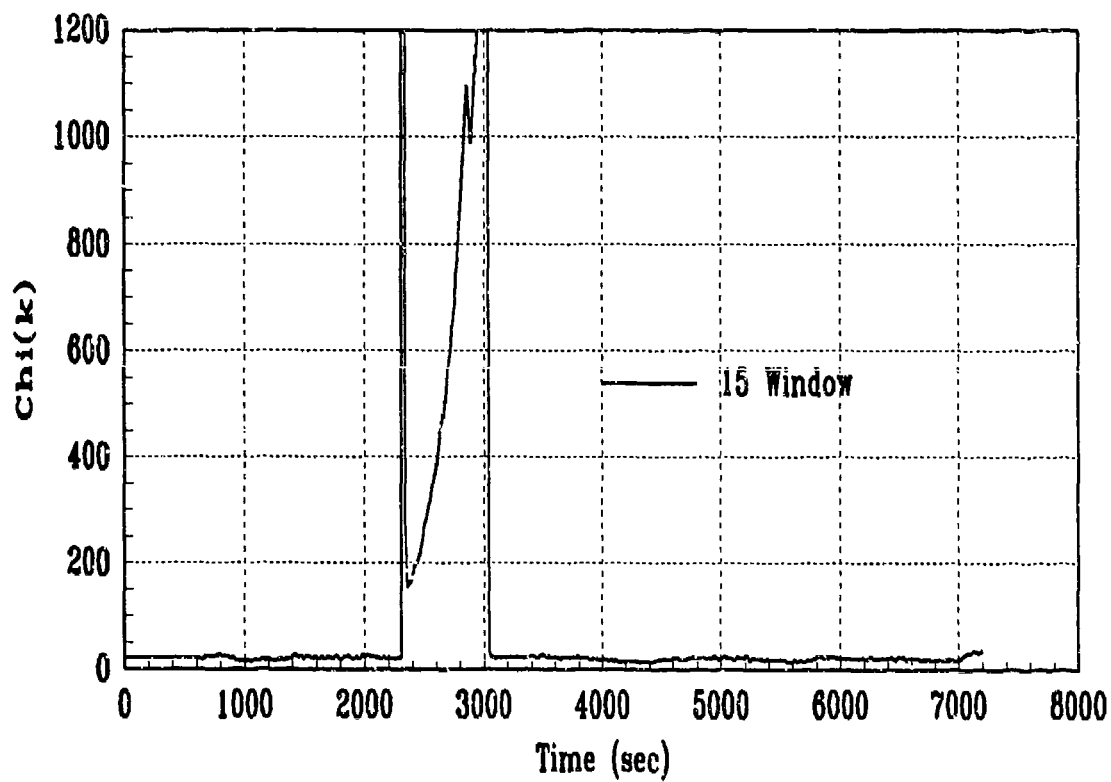
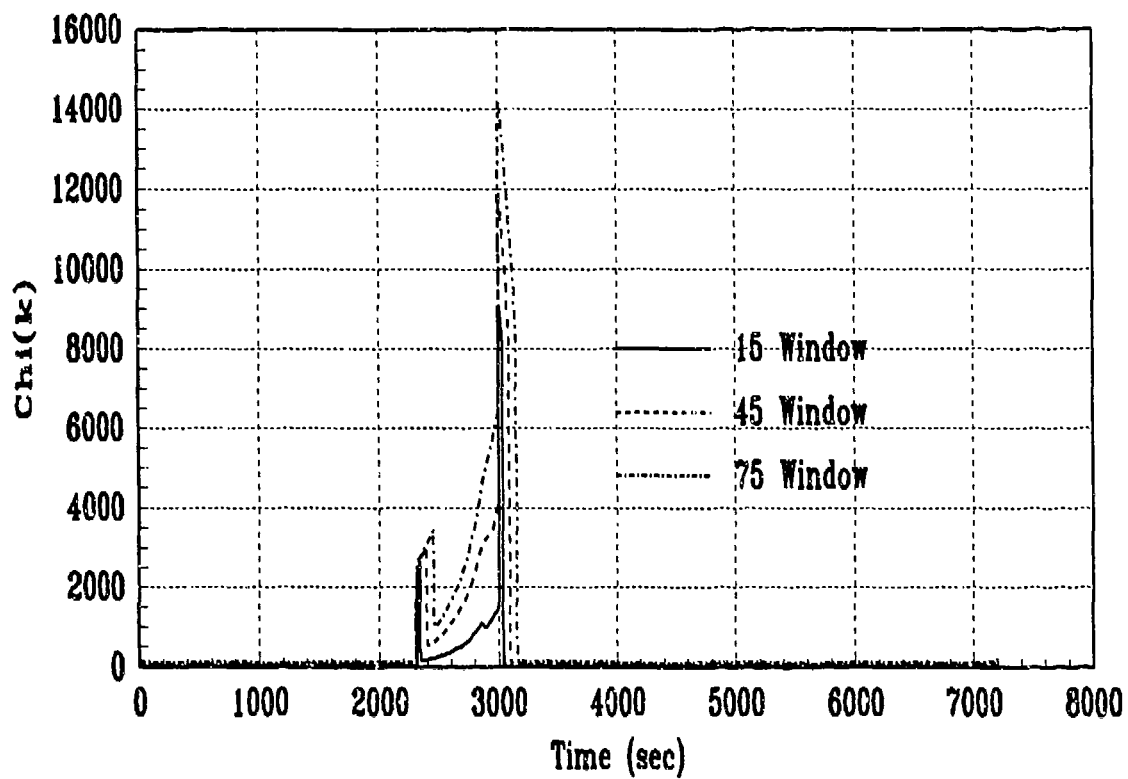
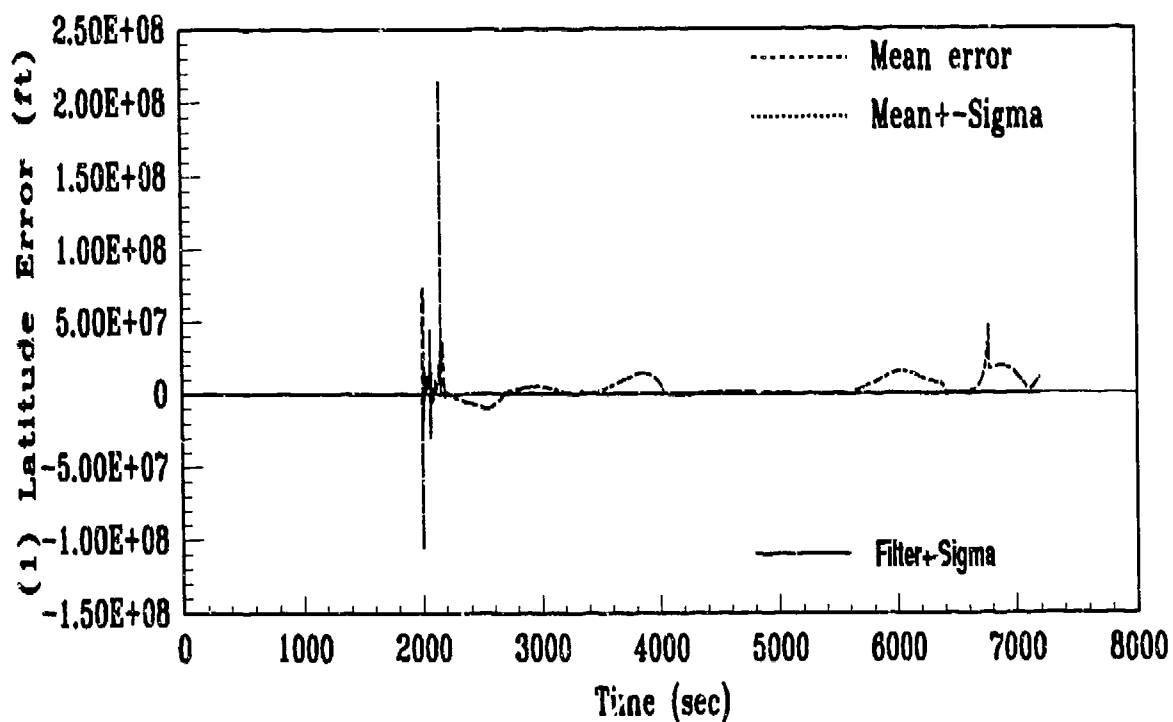


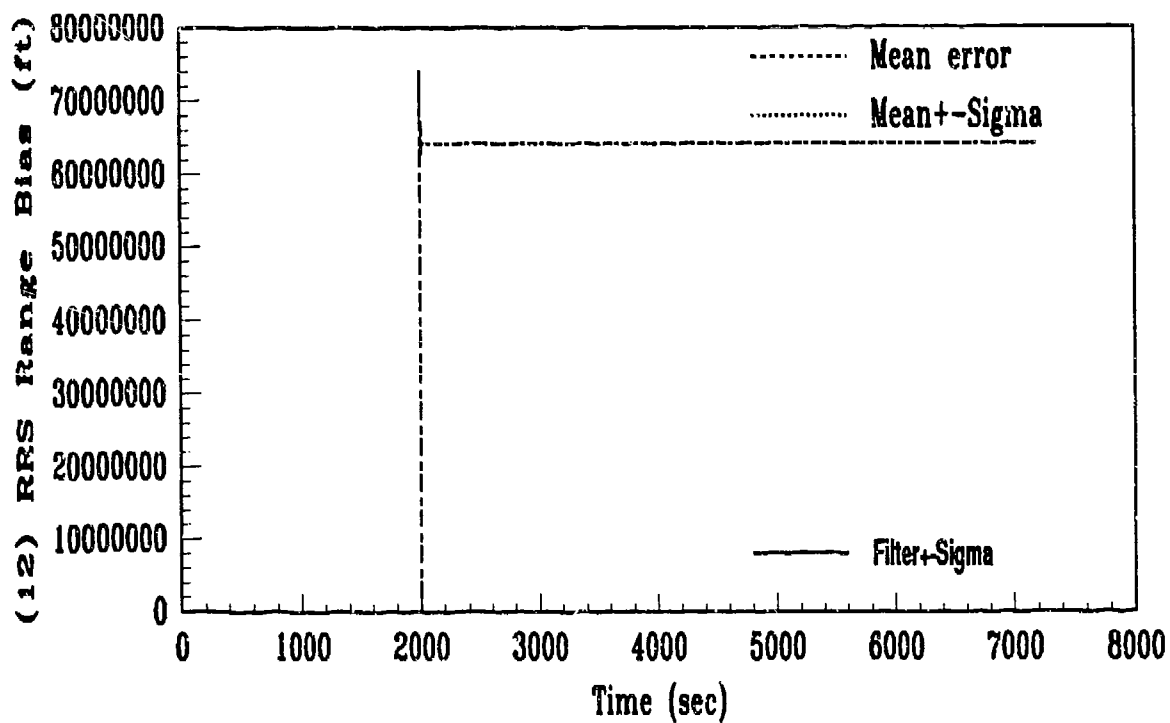
Figure G.11. CHI: Run 9, Ramp=1T

Appendix H. *GPS Failure Plots*

Selected state and residual plots are shown to illustrate the complete degradation of the NRS filter with the loss of GPS signals.

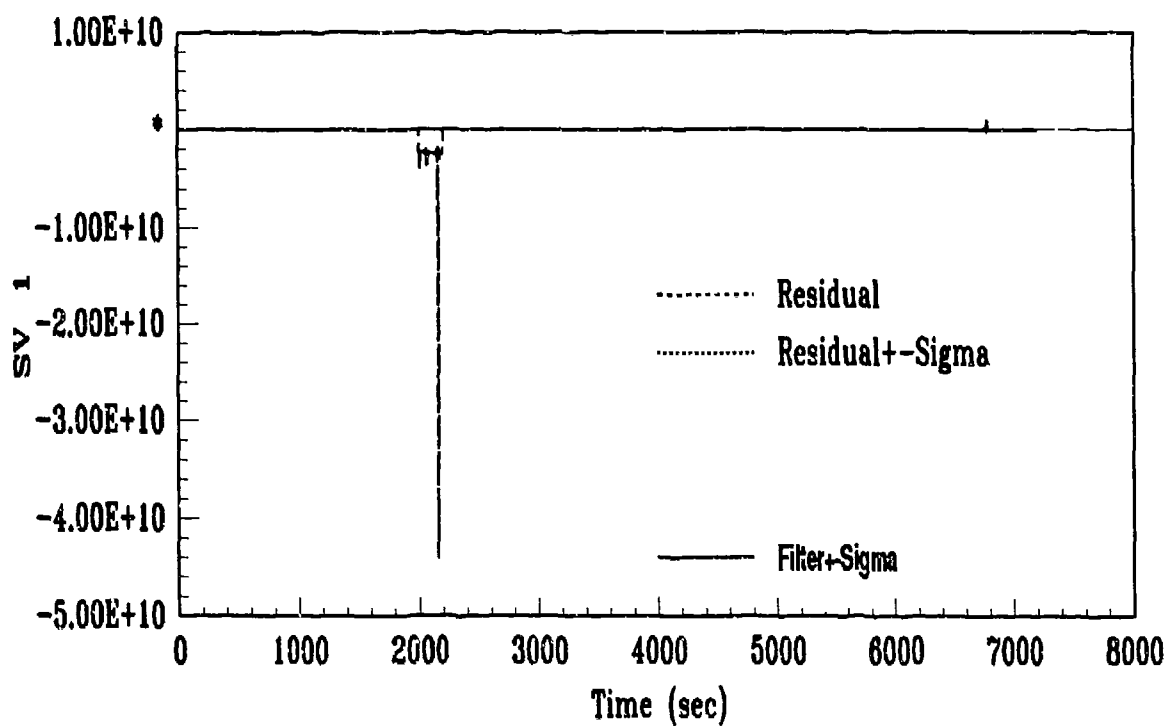


(a)

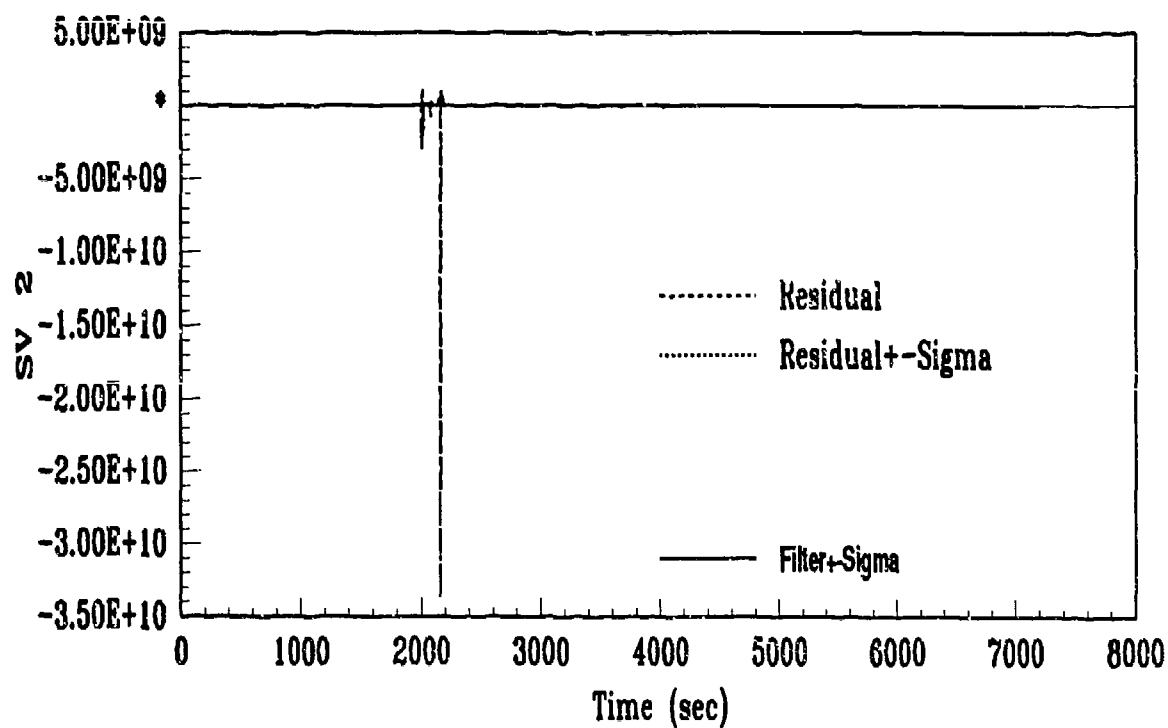


(b)

Figure H.1. State Plots: Run 10, GPS Fail

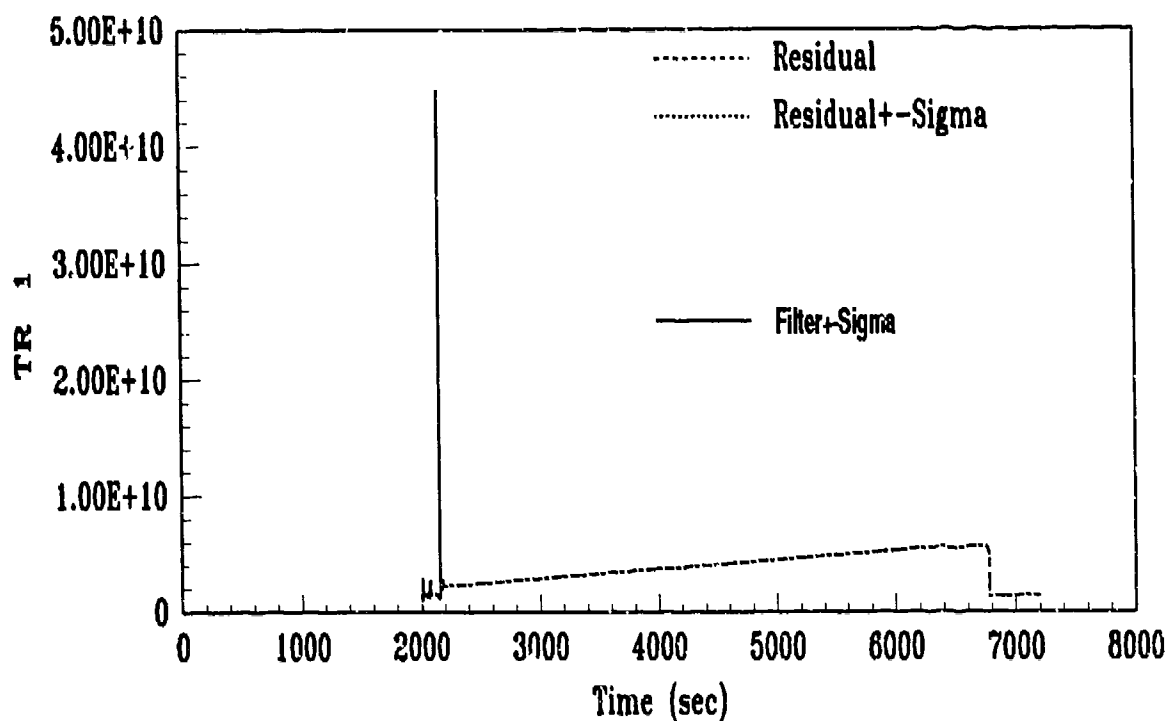


(a)

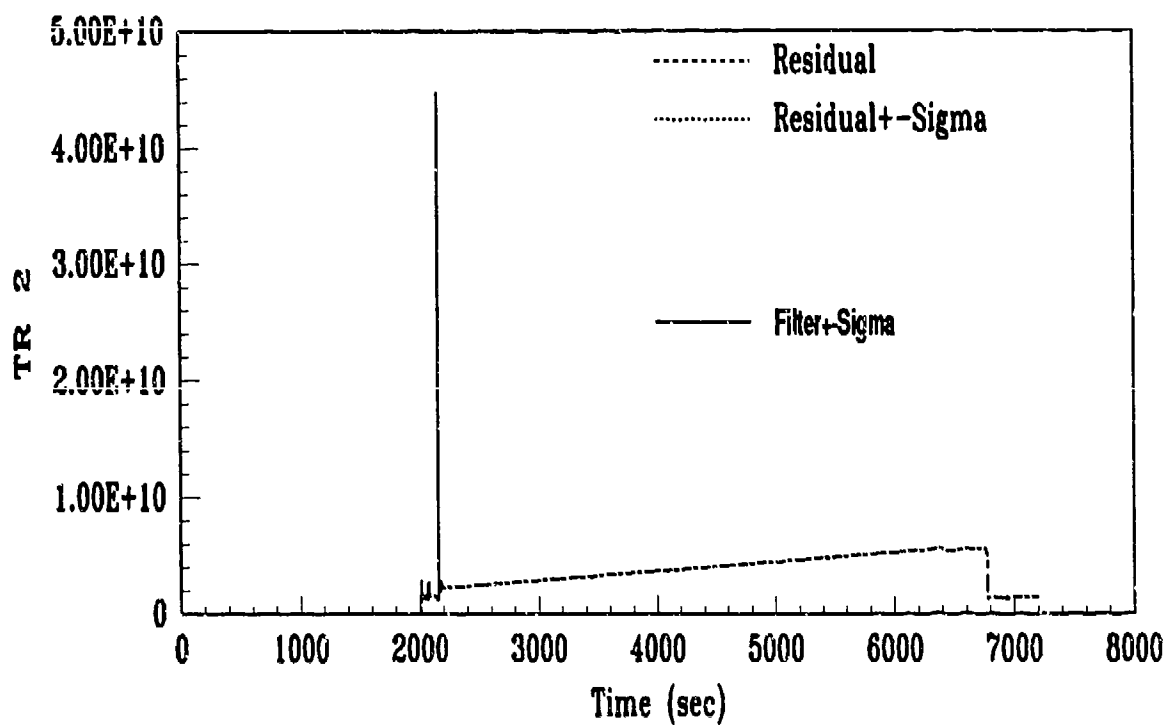


(b)

Figure H.2. Residual Plots: Run 10, GPS Fail



(a)



(b)

Figure H.3. Residual Plots: Run 10, GPS Fail

Appendix I. *Software*

The routines shown in this appendix were run on Matrix_x (8) with the data provided by MSOFE (22) and stored in Matrix_x compatible data files. The theory presented in Chapter II is implemented with documentation to assist the reader.

```
// This is a MatrixX macro that performs a GLR test with
// n = 1 for all t. The routine "lout" is called to get the
// the time-varying elements of F, H, gain, cov, & gamma stored
// in data files fout, hout, k2-k11, and uout.
// The F matrix is discretized and windowing is applied.
// Most of the variable names apply to the theory presented
// in the thesis in Chapter II for the GLR test. Only 10 measurements
// are included: SV1-4 and TR1-6 with the baro and velocity aiding
// left out as discussed in the thesis. NOTE: This routine takes
// between 30-60 minutes to run depending on the machine...be patient
//
glr="exec('glr')"; //set up macro
lout="exec('lout')"; //set up macro
//load data - NOTE: these are commented out as they loaded in already
//load 'uout.mxd',load 'hout.mxd',load 'fout.mxd',...
//load 'k2out.mxd',load 'k3out.mxd',load 'k4out.mxd',...
//load 'k5out.mxd',load 'k6out.mxd',load 'k7out.mxd',...
//load 'k8out.mxd',load 'k9out.mxd',load 'k10out.mxd',load 'k11out.mxd';
//load 'simtru.mxd';
//
//initialize constant variables
//
ns=15; //number of states
simtime=3599; //length of simulation=3599
dt=2; //sample rate
ws=10; //set window size
d=[1;0;0;0;0;0;0;0;0;0;0;0;0;0;0]; //failure matrix
n=1; //failure step function, assumed=1 for all time
yold=[0;0;0;0;0;0;0;0;0;0;0;0;0;0;0]; //starts at 0
b=0*ones(15); //driving terms matrix, not really used=0, but
// needed for the discretize command
```

```

dd=0*ones(10,15); //D matrix not really used=0, but needed
// for the discretize command
h=0*ones(10,15); //set up H-matrix
f=0*ones(15,15); //set up F-matrix
cov=0*ones(10,10); //set up residual Cov matrix
//
//constant R values
//
h(1:4,14)=[-1;-1;-1;-1];
h(5:10,12)=[-1;-1;-1;-1;-1;-1];
//
//constant F values
//
f(9,10)=3.0668051539128074D-6;
f(10,9)=1;
f(14,15)=1;
f(11,11)=-1/600;
f(9,11)=.0004;
f(10,11)=.03;
s=0;... //s for summing
c=0;... //c for summing
//
//Initialize the window
//
for k=2:ws-1,...
    kt=k,... //set index value for lout routine
    ]lout[;... //get F,H,gain(K),cov, & gamma
    ss=[f b;h dd];... //form state space matrix
    sd=discretize(ss,ns,dt,'ztransform');... //discretize the model
    [phi,bd,cd]=split(sd,ns);... //split into individual matrices
    covi=inv(cov);... //get residual cov-inverse
    ynew=phi*(eye(15)-gain*h)*yold-phi*gain*d*n;... //get y
    yold=ynew;... //store recursive value
    m=h*ynew+d;... //get m
    sterm(k)=n'*covi*gamma;... //find summation term for s
    s=s+sterm(k);... //add sterms to get s
    cterm(k)=m'*covi*m;... //find summation term for c

```

```

    c=c+cterm(k);... //add cterms to get c
    l(k)=(s/c)*s;... //MLE l(k)
end,
//
//Slide the window
//
for k=ws:simtime,...
    kt=k;... //get new value in window
    [lout[;... //get F,l,gain(K),cov, & gamma
    ss=[f b;h dd];... //form state space matrix
    sd=discretize(ss,ns,dt,'ztransform');... //discretize the model
    [phi,bd,cd]=split(sd,ns);... //split into individual matrices
    covi=inv(cov);... //get residual cov-inverse
    ynew=phi*(eye(15)-gain*h)*yold-phi*gain*d*n;... //get y
    yold=ynew;... //keep recursive value
    m=h*ynew+d;... //get m
    sterm(k)=m'*covi*gamma;... //find summation term for s
    s=s+sterm(k)-sterm(k-ws+1);... //add sterms to get s
    cterm(k)=m'*covi*m;... //find summation term for c
    c=c+cterm(k)-cterm(k-ws+1);... //add cterms to get c
    l(k)=(s/c)*s;... //MLE l(k)
end,
//
//Plot results
//
for i=1:simtime,...
    time(i)=t(2*i,1);... //keep time variable only from simtru.mxd data
end,
plot(time,l,'repo7/xlabel/Time (sec)/ylabel/MLE, L(k)/upper')

\\doorbell //indicates end of program

*****

```

```

//Routine "lout": Load in run data for post processing...
//called by macro "glr"...
//...
//Variable H, F, & K values...
//...
h(1,1)=hout(kt,1);... //H-Matrix
h(1,2)=hout(kt,2);...
h(1,10)=hout(kt,3);...
h(2,1)=hout(kt,4);...
h(2,2)=hout(kt,5);...
h(2,10)=hout(kt,6);...
h(3,1)=hout(kt,7);...
h(3,2)=hout(kt,8);...
h(3,10)=hout(kt,9);...
h(4,1)=hout(kt,10);...
h(4,2)=hout(kt,11);...
h(4,10)=hout(kt,12);...
h(5,1)=hout(kt,13);...
h(5,2)=hout(kt,14);...
h(5,10)=hout(kt,15);...
h(6,1)=hout(kt,16);...
h(6,2)=hout(kt,17);...
h(6,10)=hout(kt,18);...
h(7,1)=hout(kt,19);...
h(7,2)=hout(kt,20);...
h(7,10)=hout(kt,21);...
h(8,1)=hout(kt,22);...
h(8,2)=hout(kt,23);...
h(8,10)=hout(kt,24);...
h(9,1)=hout(kt,25);...
h(9,2)=hout(kt,26);...
h(9,10)=hout(kt,27);...
h(10,1)=hout(kt,28);...
h(10,2)=hout(kt,29);...
h(10,10)=hout(kt,30);...
f(1,3)=fout(kt,1);... //F-Matrix

```



```

f(1,8)=fout(kt,2);...
f(2,3)=fout(kt,3);...
f(2,7)=fout(kt,4);...
f(3,1)=fout(kt,5);...
f(3,2)=fout(kt,6);...
f(4,2)=fout(kt,7);...
f(4,3)=fout(kt,8);...
f(4,5)=fout(kt,9);...
f(4,6)=fout(kt,10);...
f(4,8)=fout(kt,11);...
f(5,1)=fout(kt,12);...
f(5,3)=fout(kt,13);...
f(5,4)=fout(kt,14);...
f(5,6)=fout(kt,15);...
f(5,7)=fout(kt,16);...
f(6,1)=fout(kt,17);...
f(6,2)=fout(kt,18);...
f(6,4)=fout(kt,19);...
f(6,5)=fout(kt,20);...
f(7,1)=fout(kt,21);...
f(7,2)=fout(kt,22);...
f(7,3)=fout(kt,23);...
f(7,5)=fout(kt,24);...
f(7,6)=fout(kt,25);...
f(7,7)=fout(kt,26);...
f(7,8)=fout(kt,27);...
f(7,9)=fout(kt,28);...
f(8,1)=fout(kt,29);...
f(8,2)=fout(kt,30);...
f(8,3)=fout(kt,31);...
f(8,4)=fout(kt,32);...
f(8,6)=fout(kt,33);...
f(8,7)=fout(kt,34);...
f(8,8)=fout(kt,35);...
f(8,9)=fout(kt,36);...
f(9,1)=fout(kt,37);...
f(9,2)=fout(kt,38);...

```

```

f(9,3)=fout(kt,39);...
f(9,4)=fout(kt,40);...
f(9,5)=fout(kt,41);...
f(9,7)=fout(kt,42);...
f(9,8)=fout(kt,43);...
gaint(1,1:15)=k2(kt,:);... //Gain transposed
gaint(2,1:15)=k3(kt,:);...
gaint(3,1:15)=k4(kt,:);...
gaint(4,1:15)=k5(kt,:);...
gaint(5,1:15)=k6(kt,:);...
gaint(6,1:15)=k7(kt,:);...
gaint(7,1:15)=k8(kt,:);...
gaint(8,1:15)=k9(kt,:);...
gaint(9,1:15)=k10(kt,:);...
gaint(10,1:15)=k11(kt,:);...
gammat(1:10)=u(kt,[3 5 7 9 11 13 15 17 19 21]);... //Residuals
gamma=gammat';... //transpose
gain=gaint';... //transpose
for j=1:10,...
    cov(j,j)=sqrt(u(kt,2*j));... //Residual Cov diagonal terms
end,...

```

```

//MatrixX routine to generate a chi-squared plot.
// Data files used are uout.mxd and simtru.mxd.
//
c="exec('chi')"; // set up macro
clear r v vv vi l n l lw // clear values from last run
//initialize values
k=0;
nm=10; //number of measurements
//
// Separate uout into r and v vectors. Where r represents
// the residuals, and v represents the residual covariances.
//
for i=3:2:22,...
    k=k+1;...
    r(:,k)=u(:,i);...
    v(:,k)=u(:,i+1);...
end
//
// Find rVr scalar for all time
//
for j=2:3599,... //window of time
    vv=0*ones(nm);... //initialize residual cov matrix
    for m=1:nm,...
        vv(m,m)=v(j,m);... //put diagonal terms in a square matrix
    end,...
    vi=inv(vv);... //Get vv-inverse
    rvr(j)=r(j,:)*vi*r(j,:);... //Get summation term
    time(j)=t(2*j,1);...//extract time variable from simtru.mxd
end
//
// Find chi(k) summation
//
chi(1)=rvr(1);
for k=2:3599,...
    chi(k)=chi(k-1)+rvr(k);...
end

```

```

n=15; //size of window
// initialize first window values
for j=1:n-1,...
    lw1(j)=chi(j);...
end
// compute remaining values over time by adding new
// values into the window and subtracting old values out
// of the window.
for j=n:3599,...
    lw1(j)=chi(j)-chi(j-n+1);...
end
// clean up first part of plot if transients are too large
for j=1:300,...
    lw1(j)=lw1(300);...
end

n=45; //size of window
// initialize first window values
for j=1:n-1,...
    lw3(j)=chi(j);...
end
// compute remaining values over time by adding new
// values into the window and subtracting old values out
// of the window.
for j=n:3599,...
    lw3(j)=chi(j)-chi(j-n+1);...
end
// clean up first part of plot if transients are too large
for j=1:300,...
    lw3(j)=lw3(300);...
end

n=75; //size of window
// initialize first window values
for j=1:n-1,...
    lw5(j)=chi(j);...

```

```

end
// compute remaining values over time by adding new
// values into the window and subtracting old values out
// of the window.
for j=n:3599,...
    lw5(j)=chi(j)-chi(j-n+1);...
end
// clean up first part of plot if transients are too large
for j=1:300,...
    lw5(j)=lw5(300);...
end

\\cuckoo // indicates end of routine

//plot results

plot(time,[lw1,lw3,lw5],'repo7/xlabel/Time (sec)/ylabel/Chi(k)/...
ymin=0/legend/15 Window|45 Window|75 Window/...
title/'')

```

Bibliography

1. Britting, Kenneth R. *Inertial Navigation Systems Analysis*. Wiley-Interscience, 1971.
2. Brock, Larry D. and George T. Schmidt. "Chapter 10 - General Questions on Kalman Filtering in Navigation Systems," *Theory and Applications of Kalman-Filtering, AGARDograph 139, NATO/AGARD, London, England* (Februray 1970).
3. Brown, R. and P. Hwang. "GPS Failure Detection by Autonomous Means within the Cockpit," *Institute of Navigation* (1986).
4. Ching, M., et al. *Performance of an Integrated INS and GPS with Jamming Detection*. Technical Report, AFIT, WPAFB, Ohio, Spring 1992.
5. Desai, M., et al. "Dual Sensor Failure Identification Using Analytic Redundancy," *AGARD Lecture Series No. 82* (1976).
6. Hanlon, Peter D. *Failure Identification Using Multiple Model Adaptive Estimation for the LAMBDA Flight Vehicle*. MS thesis, unpublished, Air Force Institute of Technology, December 1992.
7. Herrera, Theodore D. *Kalman Filter Tracking of a Reflective Target Using Forward Looking Infrared Measurements and Doppler Returns*. MS thesis, Air Force Institute of Technology, December 1991.
8. Integrated Systems Inc, 2500 Mission College Boulevard, Santa Clara, California 95054-1215. Matrix_x, August 1988. Version 1.2/1.3.
9. Knudsen, L. *Performance Accuracy (Truth Model/Error Budget) Analysis for the LN-93 Inertial Navigation System Inertial Navigation Unit*. Technical Report, 5500 Canoga Avenue, Woodland Hills, California 91365: Litton Guidance and Control Systems, January 1985. DID No. DI-S-21433 B/T: CDRL No. 1002.
10. Lewantowicz, Z. H. and D. W. Keen. "Graceful Degradation of GPS/INS Performance With Fewer Than Four Satellites," *The Institute of Navigation, National Technical Meeting*, 269-275 (Jan 1991).
11. Lin, Ching-Fang. *Modern Navigation, Guidance, and Control Processing*. Prentice-Hall, Inc., 1991.
12. Martin, E. H. "GPS User Equipment Error Models," *The Institute of Navigation, Volume I*:109-118 (1980).
13. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, I*. Academic Press, Inc., 1979.
14. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, II*. Academic Press, Inc., 1982.
15. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, III*. Academic Press, Inc., 1982.
16. Maybeck, Peter S. and Donald L. Pogoda. "Multiple Model Adaptive Controller for the STOL F-15 with Sensor/Actuator Failures," *Proceedings of the 28th Conference on Decision and Control*, 1566-1572 (December 1989).

17. Maybeck, Peter S. and Richard D. Stevens. "Reconfigurable Flight Control Via Multiple Model Adaptive Control Methods," *Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, Ohio*, 441-448 (May 1992).
18. Mehra, R. K. "Optimal Input Signals for Parameter Estimation in Dynamic Systems--Survey and New Results," *IEEE Trans. Automat. Control* **AC-19** (6), 753-768 (1974).
19. Mehra, R. K. "Optimal Inputs for Linear System Identification," *IEEE Trans. Automat. Control* **AC-19** (3), 192-200 (1974).
20. Menke, Timothy E. and Peter S. Maybeck. "Multiple Model Adaptive Estimation Applied to the VISTA F-16 Flight Control System with Actuator and Sensor Failures," *IEEE Trans. on Aerospace and Electronic Systems*, **AES-27**, No. 3:470-480 (May 1991).
21. Musick, Stanton H. *PROFGEN - A Computer Program for Generating Flight Profiles*. Technical Report, Air Force Avionics Laboratory, WPAFB, Ohio, November 1976. AFAL-TR-76-247, DTIC ADA034993.
22. Musick, Stanton H. *MSOFFE - Multimode Simulation for Optimal Filter Evaluation*. Technical Report, Air Force Avionics Laboratory, WPAFB, Ohio, October 1980. AFWAL-TR-88-1138.
23. Negast, William Joseph. *Incorporation of Differential Global Positioning System Measurements Using an Extended Kalman Filter for Improved Reference System Performance*. MS thesis, Air Force Institute of Technology, December 1991.
24. Ritland, John T. "Impact of Inertial System Quality on GPS-Inertial Performance in a Jamming Environment," *American Institute of Aeronautics and Astronautics*, 1459-1467 (1987).
25. Robert Nelson Riggins, Jr. *Detection and Isolation of Plant Failures in Dynamic Systems*. PhD dissertation, University of Michigan, 1991.
26. Snodgrass, Faron Britt. *Continued Development and Analysis of a New Extended Kalman Filter for the Completely Integrated Reference Instrumentation System (CIRIS)*. MS thesis, Air Force Institute of Technology, March 1990.
27. Stacey, Richard D. *A Navigation Reference System Using Global Positioning System and Transponder Aiding*. MS thesis, Air Force Institute of Technology, March 1991.
28. Stratton, Gregory L. *Actuator and Sensor Failure Identification Using a Multiple Model Adaptive Technique for the VISTA/F-16*. MS thesis, Air Force Institute of Technology, December 1991.
29. Van Trees, H. L. *Detection, Estimation and Modulation Theory*. Wiley and Sons, 1968.
30. Willsky, A. and H. Jones. "A Generalized Likelihood Ratio Approach to State Estimation in Linear Systems Subject to Abrupt Changes," *The Analytic Sciences Corporation, Reading, Mass., under USAF Contract No. F04701-74-C-0095* (1976).
31. Willsky, Alan S. "A Survey of Design Methods for Failure Detection in Dynamic Systems," *Automatica*, 601-611 (1976).
32. Zarrop, M. B. and G. C. Goodwin. "Comments on Optimal Inputs for System Identification," *IEEE Trans. Automat. Control* **AC-20** (2), 299-300 (1975).

Vita

Captain Juan Roberto Vasquez was born 1 June 1965 on Chanute AFB, Rantoul, Illinois. He has been in the Air Force since birth but did not enter active duty until 1988 after being commissioned through the Reserve Officer Training Program (ROTC) at Oklahoma State University (OSU). His life as a military dependent has sent him from England to Oklahoma where he graduated from Choctaw High School in 1983. An Air Force ROTC scholarship aided significantly in his pursuit of a Bachelor of Science Degree in Electrical and Computer Engineering from OSU in 1987. His first assignment sent him back home to Chanute AFB for technical training school to become an aircraft maintenance officer. Six months later he was sent to Kirtland AFB, NM to begin "pounding the flightline" as a supervisor. He ended this tour with a deployment to Rhein-Main, GE during Operation Desert Shield/Storm where he was the Officer In Charge of the 435th Aircraft Maintenance Unit. From June 1991 to December 1992 he attended the Air Force Institute of Technology (AFIT) where he received a Master of Science Degree in Electrical Engineering with emphasis in navigation and control systems. Following AFIT, he was married to his lovely wife, Angela, and was assigned to Space Division, Los Angeles AFB, CA, where he is currently working.

Permanent address: 18203 Ridgewood Ave
Choctaw, Oklahoma 73020

Form Approved
OMB No 0-04 0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any aspect of this collection of information, including suggestions for reducing the burden, to Washington Headquarters Service, Paperwork Project (0191-3267), Washington, DC 20503.

NSN 7540-01-280-5500

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to *stay within the lines* to meet optical scanning requirements.

Block 1. Agency Use Only (Leave blank).

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract	PR - Project
G - Grant	TA - Task
PE - Program Element	WU - Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. (If known)

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."

DOE - See authorities.

NASA - See Handbook NHB 2200.2.

NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.

DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.

NASA - Leave blank.

NTIS - Leave blank.

Block 13. Abstract. Include a brief (Maximum 200 words) factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code (NTIS only).

Blocks 17 - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.